INSTITUTO FEDERAL DO ESPÍRITO SANTO
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO APLICADA

MESSIAS GOMES DA SILVA

RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE EDITAIS DE COMPRAS UTILIZANDO APRENDIZADO PROFUNDO

Serra
2022

MESSIAS GOMES DA SILVA

RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE EDITAIS DE COMPRAS UTILIZANDO APRENDIZADO PROFUNDO

Dissertação apresentada ao Programa de Pós-Graduação em Computação Aplicada do Instituto Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Computação Aplicada.

Orientador: Prof. Dr. Hilário Tomaz Alves de Oliveira

Serra
2022

Dados Internacionais de Catalogação na Publicação (CIP)

S586r Silva, Messias Gomes da
2022 Reconhecimento de entidades nomeadas em documentos de editais de compras utilizando aprendizado profundo / Messias Gomes da Silva. - 2022.
78 f.; il.; 30 cm

Orientador: Prof. Dr. Hilário Tomaz Alves de Oliveira.
Dissertação (mestrado) - Instituto Federal do Espírito Santo, Programa de Pós-graduação em Computação Aplicada, 2022.

1. Processamento de linguagem natural (Computação). 2. Redes neurais (Computação). 3. Licitação pública. 4. Processamento de textos (Computação). I. Oliveira, Hilário Tomaz Alves de. II. Instituto Federal do Espírito Santo. III. Título.

CDD 006.32

Bibliotecária Rogeria Gomes Belchior - CRB6/ES 417

MESSIAS GOMES DA SILVA

RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE EDITAIS DE COMPRAS UTILIZANDO APRENDIZADO PROFUNDO

Dissertação apresentada ao Programa de Pós-Graduação em Computação Aplicada do Instituto Federal do Espírito Santo, como requisito parcial para obtenção do título de Mestre em Computação Aplicada.

Aprovado em 9 de dezembro de 2022.

COMISSÃO EXAMINADORA

Prof. Dr. Hilário Tomaz Alves de Oliveira
Instituto Federal do Espírito Santo
Orientador

Profa. Dra. Kelly Assis de Souza Gazolli
Instituto Federal do Espírito Santo

Prof. Dr. Fidelis Zanetti de Castro
Instituto Federal do Espírito Santo

Emitido em 09/12/2022

FOLHA DE APROVAÇÃO-TCC Nº 7/2022 - CMPCA (11.02.32.01.07.08)

(Nº do Protocolo: NÃO PROTOCOLADO)

(Assinado digitalmente em 13/12/2022 16:20)
FIDELIS ZANETTI DE CASTRO
PROFESSOR DO ENSINO BASICO TECNICO E TECNOLOGICO
SER-CGEN (11.02.32.01.08.02)
Matrícula: 1527537

(Assinado digitalmente em 13/12/2022 17:22)
HILARIO TOMAZ ALVES DE OLIVEIRA
PROFESSOR DO ENSINO BASICO TECNICO E TECNOLOGICO
SER-CGEN (11.02.32.01.08.02)
Matrícula: 3132542

(Assinado digitalmente em 14/12/2022 10:10)
KELLY ASSIS DE SOUZA GAZOLLI
PROFESSOR DO ENSINO BASICO TECNICO E TECNOLOGICO
SER-CGEN (11.02.32.01.08.02)
Matrícula: 1344568

Para verificar a autenticidade deste documento entre em https://sipac.ifes.edu.br/documentos/ informando seu número: **7**, ano: **2022**, tipo: **FOLHA DE APROVAÇÃO-TCC**, data de emissão: **13/12/2022** e o código de verificação: **608963bd9f**

Dedico este trabalho a todos os professores cuja luta para difundir o conhecimento é árdua e incessante.

AGRADECIMENTOS

Agradeço, primeiramente, à Deus por todas as oportunidades que ele vem colocando à minha disposição. Poder cursar um mestrado foi mais uma delas.

Agradeço também ao Prof. Dr. Hilário Tomaz, meu orientador, pela paciência e dedicação que ele teve na orientação deste trabalho de dissertação durante as diversas reuniões e aulas. Suas instruções e ensinamentos foram essenciais para o êxito deste trabalho.

Meus agradecimentos aos professores: Dr. Filipe Mutz, Dra. Karin Komati, Dr. Francisco Boldt e Dr. Jefferson Andrade, por terem ministrado, com muita dedicação e carinho, as aulas das disciplinas do curso de Mestrado.

Também agradeço à Profa. Dra. Kelly Assis e ao Prof. Dr. Fidelis Zanetti por terem aceitado o convite para fazerem parte da banca examinadora, pelas sugestões e contribuições dadas.

Não posso esquecer de agradecer a todos os colegas de turma pelo apoio e ajuda dados através das discussões sobre os conteúdos das disciplinas.

Agradeço ao IFES por ter me acolhido durante boa parte da minha formação acadêmica: Curso Técnico em Informática, Tecnólogo em Redes de Computadores e Mestrado em Computação Aplicada.

Quero agradecer aos meus chefes: Marcianne Ribeiro, Flávio Milanesi e Giuliano Rodrigues, pelo apoio que me deram durante essa jornada.

Agradeço à minha colega de trabalho Jussara Teixeira, pela ajuda para obtenção dos editais de licitações do Espírito Santo.

Muitíssimo obrigado ao grupo de anotadores que, durante 27 semanas, realizou o importante trabalho de anotação das entidades para confecção dos *corpora* utilizados nos experimentos. Além de mim e do Prof. Hilário, participaram do grupo: Charles Santos, Denis Wasem e Matheus Santiago. Agradeço especialmente aos três pelo tempo dedicado a essa atividade.

Agradeço à minha família, principalmente à minha filha Jéssica, por ter me apoiado e entendido que foi preciso me ausentar em alguns momentos para dedicar-me à produção deste trabalho de dissertação.

Enfim, agradeço a todos que, mesmo indiretamente, contribuíram de alguma forma para produção deste trabalho.

Darkness cannot drive out darkness: only light can do that. Hate cannot drive out hate: only love can do that. (Martin Luther King, Jr.)

RESUMO

Desde o surgimento do termo Entidade Nomeada (EN) em 1996, a tarefa Reconhecimento de Entidades Nomeadas (REN) ganhou grande interesse da comunidade de Processamento de Linguagem Natural (PLN). A tarefa de REN consiste em identificar e classificar automaticamente entidades de interesse, como nome de pessoas, lugares, organizações, entre outros, a partir de documentos textuais. Ao aplicar sistemas de REN em um domínio específico, a complexidade da tarefa é aumentada e, geralmente, há a necessidade da construção ou utilização de um *corpus* específico para esse domínio. Neste trabalho, investigou-se a aplicação de modelos de redes neurais profundas para tarefa de REN visando identificar entidades de interesse em editais de licitações de compras escritos em português do Brasil. Inicialmente, foram avaliadas arquiteturas baseadas em redes neurais profundas, especialmente usando uma *Bidirectional Long Short-Term Memory* (BiLSTM) com o algoritmo *Conditional Random Field* (CRF), chamada de BiLSTM-CRF, em conjunto com representações baseadas em *word embeddings* tradicionais (Glove e Word2vec) e contextuais (BERT, Elmo e Flair). Os modelos de *embeddings* foram avaliados isoladamente e de forma combinada. As arquiteturas foram avaliadas em diversos experimentos utilizando os *corpora* do Harem (Total e Seletivo), Paramopama e LeNER. Com base nos resultados experimentais obtidos, pode-se concluir que a BiLSTM-CRF $BERT_{Large}$-W2V, que foi a arquitetura construída com base na combinação dos *embeddings* extraídos do $BERT_{Large}$ e Word2vec, obteve os melhores resultados nos experimentos realizados e apresentou desempenho competitivo com outros sistemas do estado da arte no *corpus* Harem (Total e Seletivo). Posteriormente, um processo de anotação foi conduzido para identificação e classificação das entidades de interesse no domínio de editais de compra. Com isso, foi realizada a construção de um *corpora* contendo documentos de editais dos estados: Espírito Santo (ES), Minas Gerais (MG), Rio de Janeiro (RJ) e São Paulo (SP). Após o processo de anotação e revisão, os *corpora* foram utilizados em uma série de experimentos realizados para escolher o melhor modelo, o qual também foi baseado em uma BiLSTM-CRF, porém em conjunto somente com os *word embeddings* Glove, Word2Vec e BERT. Com base nos resultados experimentais obtidos, concluiu-se que o melhor modelo foi alcançado através de uma arquitetura usando $BERT_{Base}$ com o *Word2vec Skip-gram* que recebeu o nome BiLSTM-CRF $BERT_{Base}$-W2V.

Palavras-chave: Reconhecimento de Entidades Nomeadas. Processamento de Linguagem Natural. Redes Neurais Profundas. *Word Embeddings*. Editais de compras. Licitações.

ABSTRACT

Since the emergence of the term Named Entity (NE) in 1996, the Named Entity Recognition (NER) task has gained great interest from the Natural Language Processing (NLP) community. The NER task aims to automatically identify and classify entities of interest, such as names of people, places, organizations, and others, from textual documents. When NER systems are applied in a specific domain, the complexity of the task is increased, and it is usually necessary to build or use a domain-specific corpus. This work investigates the application of deep neural network models for the NER task seeking to identify entities of interest in procurement bid notices written in Brazilian Portuguese. Initially, architectures based on deep neural networks were evaluated, especially using a Bidirectional Long Short-Term Memory (BiLSTM) with the Conditional Random Field (CRF) algorithm, called BiLSTM-CRF, jointly with representations based on traditional (Glove and Word2vec) and contextual (BERT, Elmo and Flair) word embeddings. The embedding models were evaluated individually and combined. The architectures were evaluated in several experiments using the corpora of Harem (Total and Selective), Paramopama and LeNER. Based on the experimental results obtained, it can be concluded that the BiLSTM-CRF $BERT_{Large}$-W2V, which was the architecture built based on the combination of embeddings extracted from $BERT_{Large}$ and Word2vec, obtained the best results in the experiments carried out and showed competitive performance with other state-of-the-art systems in the Harem corpus (Total and Selective). Afterwards, an annotation process was carried out to identify and classify entities of interest in the domain of public procurement bid notices. Therefore, four corpora were built containing public notice documents for the states of Espírito Santo (ES), Minas Gerais (MG), Rio de Janeiro (RJ) and São Paulo (SP). After the annotation and review process, the corpora were used in new experiments to identify the best model, which was also based on a BiLSTM-CRF, but in conjunction only with the word embeddings Glove, Word2Vec and BERT. The experimental results demonstrated that the best model was achieved by the architecture combining the $BERT_{Base}$ with Word2vec Skip-gram, called BiLSTM-CRF $BERT_{Base}$-W2V.

Keywords: Named Entity Recognition. Natural Language Processing. Deep Neural Networks. Word Embeddings. Procurement Bid Notices. Bids.

LISTA DE FIGURAS

Figura 1 – Exemplos de entidades nomeadas reconhecidas usando a ferramenta *SpaCy*. 21
Figura 2 – Estrutura de uma classificação sequencial usando o algoritmo CRF. . . 25
Figura 3 – Arquitetura de uma rede LSTM padrão. 37
Figura 4 – Arquitetura BiLSTM-CRF. 39
Figura 5 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Large}$-W2V no Harem Total. 46
Figura 6 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Large}$-W2V no Harem Seletivo. 47
Figura 7 – Exemplo de parte do conteúdo jurídico de um edital. 50
Figura 8 – Exemplo de parte do conteúdo técnico de um edital. 50
Figura 9 – Principais atividades executadas durante o processo de criação dos *corpora* de editais. 51
Figura 10 – Exemplo de entidades anotadas nos *corpora* utilizando o esquema IOB e o formato CONLL. 54
Figura 11 – Exemplo da tela de anotação das entidades usando o Doccano. 56
Figura 12 – Estrutura de um arquivo JSONL gerado usando a ferramenta Doccano. 57
Figura 13 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Base}$-W2V no *corpus* do ES. 63
Figura 14 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Base}$-W2V no *corpus* de MG. 64
Figura 15 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Base}$-W2V no *corpus* do RJ. 65
Figura 16 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Base}$-W2V no *corpus* de SP. 66

LISTA DE TABELAS

Tabela 1 – Comparativo entre valores das receitas e pregões no estado do Espírito Santo (ES). 17
Tabela 2 – Estatísticas descritivas dos *corpora* adotados. 42
Tabela 3 – Resultados dos experimentos (%) das combinações dos *embeddings* avaliados com base na Medida-F. O maior resultado em cada *corpus* está destacado em negrito. 44
Tabela 4 – Resultados dos experimentos da arquitetura que apresentou o melhor desempenho (BiLSTM-CRF $BERT_{Large}$-W2V) avaliados com base na Precisão (%), Cobertura (%) e Medida-F por tipo de entidade nos *corpora* Harem Total e Seletivo. Os maiores resultados em cada *corpus* estão destacados em negrito. 45
Tabela 5 – Comparação dos resultados (Precisão, Cobertura e F1 em %) com outros sistemas de REN da literatura. O maior resultado em cada medição e *corpus* está destacado em negrito. 48
Tabela 6 – Quantitativo de editais coletados por estado e por ano. 52
Tabela 7 – Estatísticas descritivas dos *corpora* por estado. 58
Tabela 8 – Tipos de etiquetas e total de entidades anotadas por estado. 59
Tabela 9 – Resultados (%) dos experimentos das combinações dos *embeddings* avaliados com base na Medida-F. Os dois melhores resultados em cada coluna estão destacados em negrito. O segundo melhor resultado também está marcado com um asterisco. 61
Tabela 10 – Resultados dos experimentos da arquitetura que apresentou o melhor desempenho (BiLSTM-CRF $BERT_{Base}$-W2V) avaliados com base na Precisão (%), Cobertura (%) e Medida-F (F1) por tipo de entidade nos *corpora* divididos por estado. Os maiores resultados em cada *corpus* estão destacados em negrito. 62
Tabela 11 – Comparação dos resultados (Precisão, Cobertura e Medida-F em %) dos testes para escolha do algoritmo que foi utilizado no modelo de classificação. O maior resultado em cada medição está destacado em negrito. 77

LISTA DE QUADROS

Quadro 1 – Comparativo entre os *word embeddings* analisados. 35
Quadro 2 – Hiperparâmetros adotados nos experimentos. 43
Quadro 3 – Parâmetros utilizados nos modelos testados. 76

LISTA DE SIGLAS

ACE – Automatic Content Extraction

AGERH – Agência Estadual de Recursos Hídricos

AM – Aprendizado de Máquina

ANPD – Autoridade Nacional de Proteção de Dados

AP – Aprendizado Profundo

ASM – Approximate String Matching

BERT – Bidirectional Encoder Representation from Transformers

BiLSTM – Bidirectional Long Short-Term Memory

CBOW – Continuous Bag Of Words

CD – Coleção Dourada

CharLM – Character-level Language Modeling

CNN – Convolutional Neural Network

CoNLL – Conference on Natural Language Learning

CRF – Conditional Random Field

DA – Data Augmentation

DL – Deep Learning

DP – Desvio Padrão

EI – Extração de Informação

EN – Entidade Nomeada

FAPES – Fundação de Amparo à Pesquisa e Inovação do Espírito Santo

GB – Gigabytes

HMM – Hidden Markov Model

IA – Inteligência Artificial

JSON – JavaScript Object Notation

LSTM – Long Short-Term Memory

MLM	– Masked-Language Modeling
ML	– Modelos de Linguagem
MLP	– Multi Layer Perceptron
MUC-6	– Message Understanding Conference
NE	– Named Entity
NER	– Named Entity Recognition
NLP	– Natural Language Processing
NSP	– Next Sentence Prediction
PDF	– Portable Document Format
PLN	– Processamento de Linguagem Natural
PL	– Projeto de Lei
REN	– Reconhecimento de Entidades Nomeadas
RI	– Recuperação de Informações
RNN	– Recurrent Neural Network
SC	– Sequência de Caracteres
SGD	– Stochastic Gradient Descent
SGDW	– Stochastic Gradient Descent With Weight Decay
SVM	– Support Vector Machines
SAT	– Sumarização Automática de Texto

SUMÁRIO

1 INTRODUÇÃO . . . 16
1.1 CONTEXTUALIZAÇÃO . . . 16
1.2 JUSTIFICATIVA . . . 17
1.3 OBJETIVOS . . . 18
1.3.1 Objetivo Geral . . . 18
1.3.2 Objetivos Específicos . . . 18
1.4 ESTRUTURA DO TRABALHO . . . 19
2 RECONHECIMENTO DE ENTIDADES NOMEADAS . . . 20
2.1 DEFINIÇÃO DA TAREFA . . . 20
2.2 ABORDAGENS PARA O REN . . . 22
2.2.1 Abordagens Baseadas em Regras . . . 22
2.2.2 Abordagens Não Supervisionadas . . . 23
2.2.3 Abordagens Supervisionadas . . . 24
2.2.4 Abordagens Baseadas em Algoritmos de Aprendizado Profundo . 26
2.3 MEDIDAS E MÉTODOS DE AVALIAÇÃO DE SISTEMAS DE REN . . 29
3 RECONHECIMENTO DE ENTIDADES NOMEADAS USANDO COMBINAÇÃO DE *EMBEDDINGS* . . . 32
3.1 *WORD EMBEDDINGS* ANALISADOS . . . 33
3.2 ARQUITETURA DO MODELO DE CLASSIFICAÇÃO ADOTADO . . . 35
3.2.1 BiLSTM-CRF . . . 35
3.2.2 Flair . . . 40
3.3 CORPORA . . . 41
3.4 EXPERIMENTOS . . . 42
3.4.1 Configuração dos Experimentos . . . 43
3.4.2 Resultados dos Experimentos . . . 43
4 RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE EDITAIS DE COMPRAS . . . 49
4.1 COLETA . . . 52
4.2 PRÉ-PROCESSAMENTO . . . 52
4.3 PROCESSO DE ANOTAÇÃO . . . 53
4.3.1 Definição do Formato e Esquema de Anotação . . . 53
4.3.2 Definição das Entidades . . . 54
4.3.3 Anotação . . . 55
4.4 ESTATÍSTICAS DOS *CORPORA* . . . 58
4.5 EXPERIMENTOS . . . 59
4.5.1 Ambiente Experimental . . . 59
4.5.2 Resultados dos Experimentos . . . 60
5 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS . . . 67

5.1 CONTRIBUIÇÕES 67
5.2 TRABALHOS FUTUROS 68
5.3 PUBLICAÇÃO DECORRENTE DO TRABALHO 68
REFERÊNCIAS 69
APÊNDICE A – CLASSIFICAÇÃO DE SUBSEÇÕES TÉCNICAS OU JURÍDICAS UTILIZANDO ALGORITMOS DE APRENDIZADO DE MÁQUINA 76

1 INTRODUÇÃO

1.1 CONTEXTUALIZAÇÃO

As aquisições de bens e contratações de serviços pela Administração Pública no Brasil são efetuadas, via de regra, por licitação. Geralmente, o processo de licitação é regulamentado pela lei Federal nº 8.666, de 21 de junho de 1993, cujo foco é garantir o princípio da isonomia, a seleção da proposta mais vantajosa, economicidade, dentre outros (BRANDÃO, 2016).

Em 1º de abril de 2021 entrou em vigor a lei Federal nº 14.133 que também estabelece normas gerais de licitação e contratação para as Administrações Públicas diretas, autárquicas e fundacionais da União, dos Estados, do Distrito Federal e dos Municípios. Atualmente, as leis 8.666/1993 e 14.133/2021 estão em vigor, mas conforme o art. 193 desta última, em abril de 2023 a lei 8.666/1993 será revogada (BRASIL, 2021).

A Constituição da República Federativa do Brasil de 1988 obriga o uso de licitação, conforme inciso XXI do Art. 37 (BRASIL, 1988). Essa obrigatoriedade abrange todos os órgãos administrativos dos Poderes Executivo, Legislativo, Judiciário, dos Tribunais de Contas e do Ministério Público, autarquias, fundações públicas, empresas públicas, sociedades de economia mista, além do chamado "terceiro setor" nos casos em que haja repasse de recursos públicos para pessoas jurídicas de direito privado – como as organizações da sociedade civil de interesse público e as organizações sociais (AMORIM, 2018). Contudo, existem casos excepcionais nos quais o uso da licitação é dispensado. Nesses casos, a Constituição faz ressalva no inciso XXI do Art. 37 e a Lei 8.666, em seu artigo 24, detalha cada caso.

O processo licitatório pode ser conduzido através de diversas modalidades: leilão, concurso, concorrência, tomada de preços, convite e pregão (UNIÃO, 2010). O pregão foi instituído pela Lei nº 10.520, de 17 de julho de 2002 e pode ser presencial ou eletrônico, sendo que, no âmbito federal, o presencial é regulamentado pelo Decreto nº 3.555, de 8 de agosto de 2000 e o eletrônico pelo Decreto nº 5.450, de 31 de maio de 2005. Ambos são destinados exclusivamente à contratação de bens e serviços comuns de qualquer valor. Os licitantes, ou seja, os possíveis fornecedores, participam através de propostas de menor preço e lances no momento em que o pregão está acontecendo.

De acordo com dados coletados dos portais da transparência do Governo Federal[1] e do Governo do Estado do Espírito Santo (ES)[2], o Governo Federal movimentou no ano de 2021, somente utilizando o pregão eletrônico, cerca de 25,31 bilhões de reais. No estado do Espírito Santo, no período de 2017 a 2021, foram 9,72 bilhões de reais. Na Tabela 1 são

[1] <https://portaldatransparencia.gov.br/licitacoes>
[2] <https://transparencia.es.gov.br/Compras>

detalhados a quantidade de pregões, o total financeiro em pregões (em bilhões de reais), o total da receita estadual (em bilhões de reais) e na última coluna a relação entre pregão e a receita estadual, ano a ano no Espírito Santo.

Tabela 1 – Comparativo entre valores das receitas e pregões no estado do Espírito Santo (ES).

Ano	**Quant.**	**Pregão (bilhões R$)**	**Receita (bilhões R$)**	**Pregão/Receita (%)**
2017	2.875	1,33	14,98	8,88
2018	3.005	1,82	16,47	11,05
2019	2.605	1,38	19,33	7,14
2020	3.022	2,39	18,65	12,81
2021	3.575	2,80	19,89	14,08

Fonte: Elaborado pelo autor (2022).

Diante do que foi exposto, é possível constatar que o processo licitatório é amparado pela Constituição, algumas leis e decretos. Constata-se também que as licitações na modalidade pregão, realizadas de forma eletrônica, estão presentes na Administração Pública federal e na Administração Pública do Estado do Espírito Santo, e movimentam uma quantia expressiva de recursos financeiros.

1.2 JUSTIFICATIVA

No contexto de um processo licitatório, o edital é um documento obrigatório para todas as modalidades de licitação. Sua função é convocar os interessados em participar da licitação, conferir publicidade ao ato, estabelecer as regras e condições que deverão ser seguidas (AMORIM, 2018).

Em geral, a elaboração e validação do edital são feitas manualmente. Esse processo pode levar à ocorrência de erros ou inadequações que podem dificultar e comprometer o processo licitatório. Tais ocorrências podem provocar a impugnação do edital, causar problemas e atrasos nas aquisições de produtos ou contratações de serviços (SILVA, 2015). Em seu trabalho, Kariyado (2016) cita que problemas relacionados com a elaboração do edital podem levar à compra de bens e contratações de serviços que são inadequados e incompatíveis com o que foi pretendido no processo licitatório, resultando em prejuízos e atrasos que comprometem a prestação do serviço público.

A eficiência das compras realizadas pela Administração Pública tem como um de seus alicerces o planejamento do processo licitatório. Isso inclui, entre outros aspectos, a elaboração do edital que norteará os encaminhamentos de propostas pelos fornecedores e o julgamento destas, no sentido de atenderem ou não ao que foi solicitado (BRANDÃO, 2016).

Técnicas de Inteligência Artificial (IA) têm sido usadas para auxiliar nas atividades administrativas de órgãos públicos em diferentes esferas, assim como relatado em alguns

trabalhos identificados na literatura (SILVA, 2012; JULIÃO, 2016; CARMO et al., 2018; GARCIA; NETO, 2018; ALLES, 2018; CASTRO, 2019; ISHIKAWA; ALENCAR, 2020). A aplicação de técnicas de Processamento de Linguagem Natural (PLN) em conjunto com algoritmos de Aprendizado de Máquina (AM) e, mais recentemente, de Aprendizado Profundo (AP), têm demandado interesse dos pesquisadores diante da grande quantidade de documentos textuais produzidos nos mais diversos níveis das organizações públicas.

Em especial, a tarefa de Reconhecimento de Entidades Nomeadas (REN) tem sido objeto de estudo em diferentes domínios, desde documentos jurídicos (CASTRO, 2019), notícias do governo (SILVA, 2012), diário oficial da união (ALLES, 2018), entre outros. A tarefa de REN tem por objetivo identificar e classificar menções de entidades de interesse, por exemplo, nomes de pessoas, organizações, lugares, entre outros, em documentos textuais. Essa tarefa serve como base para várias outras aplicações de PLN como Sumarização Automática de Texto (SAT) (OLIVEIRA et al., 2016), Recuperação de Informações (RI) (LIU et al., 2015), Extração de Informação (EI) (JIANG, 2012), entre outras.

Aplicar REN em documentos de editais de licitações de compra, possibilitará a realização de buscas por entidades de interesse que poderão ser utilizadas pelas pessoas que participam do processo de confecção desses documentos, com o intuito de verificar se há erros para corrigir e validar a redação. Dessa forma, a etapa de REN pode servir como base para futuras aplicações que possam otimizar o processo de escrita e análise automática de editais. Neste contexto, esta pesquisa tem por objetivo avaliar a aplicação de diferentes arquiteturas baseadas em redes neurais profundas para a tarefa de REN em documentos de editais de licitações de compras escritos em português do Brasil.

1.3 OBJETIVOS

Nesta seção são descritos o objetivo geral e os objetivos específicos que deverão ser alcançados com a produção deste trabalho.

1.3.1 Objetivo Geral

O objetivo geral deste trabalho é avaliar a aplicação de diferentes arquiteturas baseadas em redes neurais profundas para o reconhecimento de entidades nomeadas em documentos de editais escritos em português do Brasil.

1.3.2 Objetivos Específicos

Para alcançar o objetivo geral definido neste trabalho, alguns objetivos específicos são elencados a seguir.

- Desenvolver um modelo de reconhecimento de entidades nomeadas baseado em redes neurais profundas;

- Implementar diferentes combinações de representações das palavras baseadas em *word embeddings* tradicionais e contextuais;

- Executar experimentos do modelo de classificação de entidades em diferentes *corpora* identificados na literatura;

- Construir *corpora* de editais de compras com entidades nomeadas de interesse anotadas;

- Realizar experimentos para avaliar o modelo de classificação de entidades nos *corpora* de editais construídos.

1.4 ESTRUTURA DO TRABALHO

Além deste capítulo introdutório, o restante do documento está organizado da seguinte forma:

No Capítulo 2 são apresentados alguns conceitos básicos sobre a tarefa de Reconhecimento de Entidades Nomeadas (REN), algumas abordagens usadas na literatura e alguns trabalhos relacionados.

No Capítulo 3 são apresentados diversos experimentos realizados para avaliar a eficiência da estratégia de combinação de *word embeddings* tradicionais e contextuais para a tarefa de REN em português em diferentes *corpora* disponíveis na literatura.

No Capítulo 4 é apresentado o processo de construção dos *corpora* de editais propostos neste trabalho que serão utilizados para treinamento e testes que apoiarão a tarefa de REN. Além disso, são apresentadas as atividades realizadas para construção dos *corpora*, que envolveu a extração dos textos dos editais de compras; geração dos arquivos que foram anotados pelo grupo de anotadores; validação dos arquivos anotados e geração dos *corpora* finais. Por fim, são apresentados e discutidos os experimentos realizados com base nos *corpora* para escolha do melhor modelo de REN.

Por último, no Capítulo 5 são apresentadas as considerações finais deste trabalho, as contribuições geradas e são delineadas algumas linhas de trabalhos futuros.

2 RECONHECIMENTO DE ENTIDADES NOMEADAS

Neste capítulo são apresentados alguns conceitos básicos sobre a tarefa de Reconhecimento de Entidades Nomeadas (REN) que são importantes para o entendimento do restante deste trabalho. Para isso, na Seção 2.1 é exposta uma definição da tarefa de REN, na Seção 2.2 são apresentadas algumas abordagens comumente usadas na literatura, bem como alguns trabalhos relacionados. Por fim, na Seção 2.3 são explicadas algumas medidas usadas para avaliar sistemas de REN.

2.1 DEFINIÇÃO DA TAREFA

O termo Entidade Nomeada (EN) foi utilizado pela primeira vez em 1995, na *Message Understanding Conference* (MUC-6) (GRISHMAN; SUNDHEIM, 1996). Desde então, a tarefa REN ganhou interesse da comunidade de Processamento de Linguagem Natural (PLN). Uma EN representa algo no mundo real, por exemplo, nomes de pessoas, empresas, organizações, lugares, entre outros. A tarefa de REN consiste em identificar e classificar automaticamente menções dessas entidades a partir de documentos textuais. Quando encontradas, as entidades devem ser classificadas em categorias semânticas pré-definidas como pessoa, localização, organização, e assim por diante (JIANG, 2012). Li et al. (2020) ressaltam que para alguns domínios, de acordo com suas particularidades, há a necessidade de utilizar categorias de entidades mais específicas como genes, proteínas, enzimas, dentre outras.

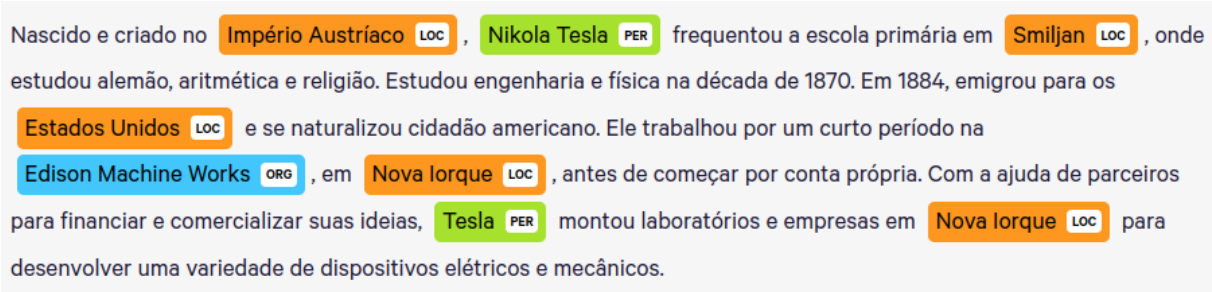
Para realizar a tarefa de REN em textos, inicialmente é necessário dividi-los em unidades menores chamadas *tokens*. Segundo Manning, Raghavan e Schütze (2009), um *token* é uma instância formada por uma sequência de caracteres, em um documento em particular, que pode ser processada. Por exemplo, se a frase *"Maria comprou um belo sapato."* for dividida em *tokens*, será gerada a seguinte lista: (*"Maria", "comprou", "um", "belo", "sapato" e "."*). Dependendo das particularidades de cada texto e da ferramenta utilizada para realizar o processo de *tokenização*, um *token* pode representar também símbolos, números, dentre outras sequências de caracteres que não formam, necessariamente, palavras válidas.

Na Figura 1 é ilustrado um exemplo de etiquetagem para a tarefa de REN produzido utilizando as ferramentas SpaCy[3] e *displaCy Named Entity Visualizer*[4]. Para isso, foi usado um fragmento de um texto da Wikipédia sobre o inventor Nikola Tesla[5]. Os seguintes tipos de entidades foram identificados: *Person* (PER), *Organization* (ORG) e *Location* (LOC). Observa-se na Figura 1 que a ferramenta etiquetou *Império Austríaco*, *Smiljan*, *Estados Unidos* e *Nova Iorque* como menções de locazilações (LOC); *Nikola Tesla* e *Tesla*

[3] <https://spacy.io/>
[4] <https://explosion.ai/demos/displacy-ent>
[5] <https://pt.wikipedia.org/wiki/Nikola_Tesla>

como pessoas (PER), e *Edison Machine Works* foi etiquetada como organização (ORG).

Figura 1 – Exemplos de entidades nomeadas reconhecidas usando a ferramenta *SpaCy*.



Fonte: Elaborado pelo autor (2022).

A tarefa de REN não é trivial, principalmente em casos que envolvem certa ambiguidade, como no reconhecimento de pessoas e organizações, pois, a depender do contexto, pode não ser possível determinar corretamente à qual classe a entidade pertence. Fonseca et al. (2015) citam dois exemplos de frases que podem envolver ambiguidade:

1. "A **Mercedes** esteve em crise nos últimos meses. A empresa afirma que ..."

2. "Hoje foi um ótimo dia para **Mercedes**."

No exemplo 1, se a relação entre os sintagmas nominais [**A Mercedes**] e [**A empresa**] for estabelecida, é possível que ajude na desambiguação da entidade **Mercedes** e assim etiquetá-la corretamente como organização e não como pessoa. Entretanto, na frase 2 a informação não é suficiente para afirmar, com certeza, que a entidade é uma organização, como no primeiro exemplo, ou uma pessoa.

Outro desafio da tarefa de REN é identificar corretamente os limites das entidades, ou seja, quando uma entidade é composta por mais de um *token*. Neste caso, é necessário identificar quando a menção da entidade começa e quando termina, por exemplo, a expressão "Joaquim José da Silva Xavier" é composta por cinco palavras e deve ser reconhecida como uma menção de uma entidade da classe pessoa. Nesse exemplo, um sistema de REN eficiente deve identificar que a entidade começa com a palavra "Joaquim" e finaliza somente com a palavra "Xavier".

A tarefa de REN é muito importante para outras aplicações de PLN, sendo amplamente utilizada em tarefas como: Recuperação de Informação (LIU et al., 2015), Extração de Informação (JIANG, 2012), Sistemas de Perguntas e Respostas (MERVIN, 2013), Sumarização Automática de Textos (BARALIS et al., 2013; OLIVEIRA et al., 2016), Tradução Automatizada de Textos (SANTY; BHATTACHARYA, 2021), entre outras.

2.2 ABORDAGENS PARA O REN

Desde a competição realizada na conferência do MUC-6 (GRISHMAN; SUNDHEIM, 1996), diversos trabalhos têm investigado a tarefa de REN, propondo e avaliando novas soluções. Os primeiros sistemas de REN eram baseados em abordagens clássicas usando regras criadas manualmente ou listas predefinidas conhecidas como *gazetteers*. Recentemente, arquiteturas baseadas em redes neurais profundas ganharam destaque devido aos resultados do estado da arte obtidos.

Existem basicamente quatro abordagens usadas na literatura para realizar as etapas de reconhecimento, etiquetagem e extração de EN, sendo elas (LI et al., 2020):

1. **Baseadas em regras** onde as entidades são extraídas utilizando um conjunto de padrões, que são geralmente definidos manualmente ou obtidos utilizando dicionários.

2. **Supervisionadas** que utilizam textos etiquetados e um conjunto de características para representar as palavras, e treinar modelos de AM para identificação e classificação das entidades.

3. **Não supervisionadas** em que não há necessidade de textos etiquetados e empregam algoritmos de agrupamento ou de similaridade para o reconhecimento das entidades.

4. Baseadas em algoritmos de **Aprendizado Profundo** que conseguem extrair a partir de um conjunto de textos etiquetados as representações necessárias para a identificação e classificação das entidades, sem a necessidade de ter explicitamente uma etapa de extração de características, assim como nas abordagens supervisionadas clássicas.

Nas subseções a seguir, são apresentados mais detalhes e alguns trabalhos relacionados de cada uma dessas quatro abordagens.

2.2.1 Abordagens Baseadas em Regras

A identificação e extração de entidades nomeadas baseada em regras é realizada através do reconhecimento de padrões no texto que indicam a presença de uma possível entidade de interesse. Em geral, esses padrões são definidos manualmente utilizando regras que determinam como identificar a entidade, e uma vez identificada, em qual tipo ela será classificada. A construção dessas regras de forma manual é muito trabalhosa e há uma dependência muito grande do domínio para o qual elas foram criadas (JIANG, 2012). De acordo com Jiang, Banchs e Li (2016), realizar a transferência dessas regras de um domínio para outro é desafiador.

Jiang (2012) descreve que uma abordagem tradicional baseada em regras para a tarefa de REN pode ser realizada aplicando as seguintes etapas: **(i)** definição do conjunto de regras. Essas regras consistem em um padrão (que pode ser definido utilizando expressões regulares) e uma ação a ser tomada (que nesse caso é em qual tipo de entidade o *token* será classificado); **(ii)** aplicação das regras criadas nos textos de interesse; e **(iii)** caso algum *token* (possível entidade) possua características que se encaixem com uma das regras do conjunto, essa é utilizada para classificá-lo.

Um importante recurso que pode ser criado para auxiliar na identificação de entidades nomeadas são dicionários ou *gazetteers*, que são listas pré-definidas de termos organizados por rótulos (Local, Pessoa, e demais rótulos). Dessa forma, cada um dos *tokens* de um texto são comparados com os termos dessas listas e caso coincidam, o *token* receberá o mesmo rótulo do termo da lista (SANTOS, 2017). Por exemplo, se o interesse é identificar menções dos estados brasileiros, pode-se definir uma lista com os nomes dos estados e utilizá-la para auxiliar a categorização.

O trabalho desenvolvido por Ferri (2016) apresenta a criação de um sistema de REN baseado em dicionários que utiliza uma árvore de prefixos para indexar os termos do dicionário e possibilitar buscas mais eficientes. Para que o sistema não deixe de reconhecer uma entidade caso haja algum erro ortográfico na palavra buscada, foi utilizada a técnica de *Approximate String Matching* (ASM) que consiste em localizar ocorrências de uma Sequência de Caracteres (SC) em outra SC maior, possibilitando que duas palavras sejam consideradas correspondentes mesmo que existam erros em uma delas ou em ambas. Como medida de similaridade para a ASM foi utilizada a distância de *Levenshtein* que mensura a distância entre duas SC com base no número de alterações necessárias em uma SC para ficar igual a outra e considerá-las correspondentes, sendo que quanto menos alterações mais semelhantes as SC são. Também foi utilizado o algoritmo de transformação *Double Metaphone* com o objetivo de comprimir as SCs, fazendo com que ocorresse uma diminuição do consumo de memória, além de eliminar alguns erros fonéticos. Após a execução de vários experimentos o autor concluiu que: **(i)** a personalização dos parâmetros do sistema pode priorizar diferentes características das entradas e dos textos, sendo possível obter melhores valores nas medidas de avaliação de Precisão e Cobertura para determinado contexto; e **(ii)** as transformações realizadas utilizando o *Double Metaphone* diminuíram o tamanho da árvore de prefixos e, juntamente com a medida de similaridade, eliminaram os erros nas palavras permitindo reconhecer um maior número de entidades.

2.2.2 Abordagens Não Supervisionadas

As abordagens não supervisionadas não necessitam de textos etiquetados, ou seja, referências que auxiliam no reconhecimento das entidades. O processo de aprendizado acontece no momento em que o algoritmo aprende a representar (ou agrupar) as entradas utilizando

medidas de similaridade que permitem processar os dados através de padrões ou tendências (PELLUCCI et al., 2011).

A técnica de clusterização ou agrupamento é utilizada para formar grupos de EN considerando as similaridades entre elas. Nessa técnica, uma palavra que consta em um documento é passada como entrada e as palavras que estão em torno dela (considerando uma janela fixa) são utilizadas para verificar a similaridade e atribuir um rótulo para a palavra de entrada baseado no rótulo da palavra vizinha mais semelhante (SILVA, 2012).

Li et al. (2019) empregaram uma abordagem de aprendizado não supervisionado para ampliar a anotação de textos de enciclopédias online, no idioma chinês, visando reduzir a necessidade de textos anotados manualmente sem impactar negativamente na precisão do modelo de REN. Os textos da Wikipédia foram divididos em dois grupos, conforme a quantidade de ligações para outros textos que podem ser utilizadas para definir se os *tokens* são entidades e a qual classe de entidade pertencem. O primeiro grupo são textos que possuem maior quantidade de ligações com outros textos e o segundo grupo, textos com poucas conexões. Foi empregada uma técnica de *Data Augmentation* (DA) utilizando o primeiro grupo para treinar um modelo do tipo BiLSTM-CRF e efetuar a etiquetagem de REN no segundo grupo para aumentar a quantidade de sentenças anotadas. Um dos *datasets* utilizados foi composto por 8000 sentenças retiradas de textos sobre geografia, que foi usado para treinar o modelo, e o outro foi o *Microsoft Research Asia Chinese Word-Segmentation Data Set* (MSRA NER), usado para avaliar os efeitos do procedimento de DA. Os autores concluíram que o modelo desenvolvido seguindo a abordagem não supervisionada obteve resultados equivalentes a abordagens supervisionadas.

2.2.3 Abordagens Supervisionadas

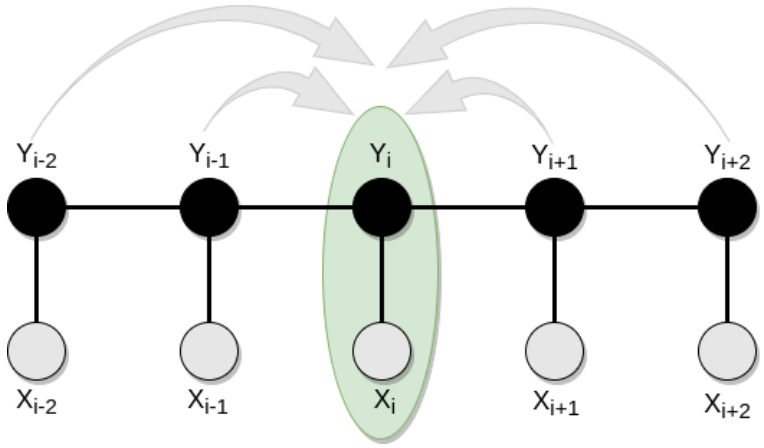
Neste tipo de abordagem são necessárias amostras dos dados etiquetados para que o algoritmo de REN extraia e aprenda uma função que defina a relação dos exemplos com as suas respectivas classes. Dessa forma, é possível inferir a quais classes os dados ainda não vistos pelo algoritmo pertencem. No contexto de PLN, as amostras de dados utilizadas nesse processo são textos que possuem as entidades devidamente anotadas (CARVALHO, 2012).

As abordagens supervisionadas podem ser realizadas em basicamente duas etapas: (i) **Treinamento**, etapa em que o modelo é submetido ao conjunto de textos com as entidades previamente anotadas com o objetivo de aprender a reconhecê-las automaticamente; e (ii) **Testes**, onde um conjunto de dados que não participaram do treinamento são classificados usando o modelo treinado e comparados com as entidades verdadeiras para computar medidas de avaliação. A tarefa de REN é considerada um problema de classificação, cuja finalidade é identificar se uma palavra pertence ou não ao conjunto de entidades de

interesse (PRIVATTO, 2020).

A tarefa de REN precisa levar em consideração o contexto no qual as palavras estão inseridas. Assim, a decisão de qual rótulo atribuir a uma palavra precisa considerar as classificações das palavras vizinhas. Nesse contexto, o algoritmo de *Conditional Random Fields* (CRF), proposto por Lafferty, McCallum e Pereira (2001), pode ser utilizado para segmentar e etiquetar sequências condicionalmente, em que os rótulos dos elementos vizinhos servem para determinar o rótulo de um elemento alvo. Essa dependência é ilustrada na Figura 2, em que o rótulo de X_i, que neste caso é Y_i, depende dos rótulos vizinhos Y_{i-1}, Y_{i-2}, Y_{i+1} e Y_{i+2} que foram atribuídos, respectivamente, aos elementos X_{i-1}, X_{i-2}, X_{i+1} e X_{i+2}. Uma sequência de entrada é dada por $X = \{X_1, X_2, X_3, ..., X_n\}$ e uma sequência de saída por $Y = \{Y_1, Y_2, Y_3, ..., Y_n\}$, onde X é a sequência de características, Y é a sequência de rótulos, X_i e Y_i representam o valor e o rótulo do i-ésimo elemento, respectivamente.

Figura 2 – Estrutura de uma classificação sequencial usando o algoritmo CRF.



Fonte: Adaptado de Lafferty, McCallum e Pereira (2001).

O modelo CRF vem sendo utilizado em vários trabalhos para execução da tarefa de REN em diversos idiomas, como: português (PIROVANI; OLIVEIRA, 2018), inglês (PANCHENDRARAJAN; AMARESAN, 2018), indiano (DAS; GARAIN, 2014), russo (ARKHIPOV; BURTSEV et al., 2017), chinês (JIN et al., 2019) e tcheco (KONKOL; KONOPÍK, 2013). Nesses trabalhos, geralmente, o CRF é usado para aprender as relações entre os *tokens* com base em características tais como se o *token* é composto somente por caracteres em maiúsculo, ou somente por caracteres em minúsculo ou uma mistura de ambos, se o *token* inicia com letra maiúscula, minúscula ou por símbolos, se o *token* é composto por símbolos, números ou letras, pela classe gramatical que o *token* pertence), dentre outras características. Além disso, o algoritmo CRF pode ser combinado com outros modelos e arquiteturas para a realização da tarefa de REN.

Silva (2012) desenvolveu um sistema de REN específico para notícias do governo brasileiro redigidas em português. Esse escopo foi definido pelo autor para investigar se existe um ganho de desempenho em detrimento da escolha de um modelo genérico para realizar a tarefa proposta. Concluiu-se que a escolha de um modelo mais específico influenciou diretamente na complexidade do problema e facilitou no aprendizado do modelo. Segundo o autor do trabalho, o método *Linear-Chain Conditional Random Field* foi escolhido porque é baseado em modelos probabilísticos e o modelo pode ser adaptado para reconhecimento de outros idiomas, pois o aprendizado é feito estatisticamente e não há necessidade de trabalhar com conceitos semânticos específicos de um idioma para que o modelo aprenda a reconhecer as entidades. O autor cita também que a metodologia supervisionada foi escolhida para conduzir o trabalho, pois permite a transferência de conhecimento utilizando uma base de treinamento e possibilita a comparação de resultados entre a base de treinamento proposta para o trabalho e a base existente da Coleção Dourada (CD).

Amaral (2013) construiu um sistema chamado NERP-CRF que utiliza o modelo CRF. O modelo proposto foi treinado e testado com base nos *corpora* referentes às Coleções Douradas do Primeiro e do Segundo HAREM. Os textos de entrada foram pré-processados em duas etapas: **(i)** cada palavra foi etiquetada com base na sua classe gramatical, que consiste em classificar as palavras em pronomes, advérbios, verbos, e outras classes gramaticais; **(ii)** separação em sentenças. Após o pré-processamento do texto foi gerado um vetor de características tais como: *tag* (etiqueta *Part-of-Speech* (POS) da palavra), *word* (a palavra em si), *prevW* (a palavra anterior), somente para citar algumas. Foram passados como entrada para o CRF, o vetor com as etiquetas das classes gramaticais, as categorias das palavras e o vetor de *features* para que o sistema aprendesse a reconhecer as entidades no texto. Depois da execução de três testes para verificação do desempenho do sistema, o mesmo foi comparado com outros sistemas que participaram do segundo HAREM, a saber, Priberam, R3M, Rembrandt, SEI-Geo e CaGE. Como resultado da comparação, a autora relata que o NERP-CRF, no segundo teste, obteve melhores resultados que os sistemas analisados para as medidas de avaliação de Precisão e Medida-F.

2.2.4 Abordagens Baseadas em Algoritmos de Aprendizado Profundo

A adoção de arquiteturas de redes neurais profundas e Modelos de Linguagem (ML) pré-treinados para extrair as características de uma linguagem automaticamente foi um avanço para a tarefa de REN (LI et al., 2020). As representações das palavras criadas servem como entrada para um classificador que gerará a rotulagem final das palavras encontradas em um texto em que se deseja extrair as entidades (NETO, 2019). Segundo Athavale et al. (2016), as Redes Neurais Recorrentes, do inglês *Recurrent Neural Network* (RNN), e suas variantes têm obtido resultados do estado da arte para a tarefa REN.

Santos e Guimarães (2015) propuseram um sistema de REN independente de idioma

chamado de CharWNN, que é uma rede neural utilizada para classificação que também captura as representações em nível de caracteres. Para isso, os autores utilizaram dois tipos de representações em *embeddings*: **(i)** baseado em palavras, onde são capturadas informações semânticas e sintáticas; e **(ii)** baseado em caracteres que capturam informações morfológicas e de formato. Os *corpora* utilizados no trabalho foram HAREM I e SPA CoNLL-2002, compostos por textos em português e espanhol, respectivamente. O sistema desenvolvido foi comparado, utilizando o *corpus* SPA CoNLL-2002, com as redes neurais WNN (com duas configurações distintas), CharNN e AdaBoost, e obteve os melhores resultados. Entretanto, na comparação usando o *corpus* HAREM I e substituindo o AdaBoost pelo ETL_{CMT}, o sistema obteve resultados encorajadores, mas não superou as outras redes em todas as medidas de avaliação.

Castro, Silva e Soares (2018) conduziram um estudo avaliando a estratégia de combinar diferentes tipos de *embeddings* no nível de caracteres e no nível de palavras para representar as palavras dos textos de entrada. Para realizar a identificação e classificação das entidades, foi adotada uma rede do tipo *Bidirectional Long Short-Term Memory* (BiLSTM) combinada com uma camada composta pelo modelo *Conditional Random Fields* (CRF). Os autores investigaram a utilização dos *embeddings* pré-treinados do FastText, Glove, Wang2Vec e Word2Vec, e adotaram os *corpora* HAREM I, como conjunto de treinamento, e o MiniHAREM, como teste. A arquitetura do sistema utilizado no estudo é composta por duas redes LSTM, uma para aprender as características utilizando *embeddings* baseados em caracteres e a outra utilizando *embeddings* baseados em palavras. Nos resultados do estudo, foi observado que: **(i)** dentre os parâmetros utilizados, os que causaram maior impacto no desempenho foram a normalização das palavras e o tipo de *embedding*; **(ii)** que o *embedding* Wang2Vec obteve o melhor desempenho em comparação com os outros; e **(iii)** a rede LSTM combinada com CRF superou os resultados obtidos pela rede CharWNN, em 5 pontos na métrica Medida-F, nos cenários total e seletivo do HAREM I.

Santos et al. (2019) avaliaram o impacto da combinação de *word embeddings* tradicionais (chamados de *shallow*) e contextuais (gerados em tempo de execução de acordo com a sentença dada como entrada para o modelo) na tarefa de REN para o idioma português. Foram feitas diversas combinações dos tipos de *embeddings* investigados para medir o impacto causado pela diversidade textual e do tamanho dos *corpora* de treinamento. Para realizar a avaliação, os autores pré-treinaram o modelo *Flair Embedding LM* utilizando um *corpus* com 4,9 bilhões de palavras – em texto puro – e combinaram com os *embeddings* Word2Vec Skip-gram. O modelo produzido recebeu o nome FlairBBP e foi usado como entrada para uma rede BiLSTM-CRF. Os testes foram realizados com base nos *corpora* do HAREM. Além disso, para treinar os *embeddings* foram usados os *corpora* BlogSet-BR, brWaC e ptwiki-20190301, e os *embeddings* FasText and Word2Vec. O melhor resultado da avaliação foi obtido com a combinação do FlairBBP com Word2Vec Skip-gram. Os

autores relataram que o experimento demonstrou que a diversidade textual e o tamanho do *corpora*, assim como o tipo de treinamento utilizado na criação dos modelos de *embeddings*, podem influenciar na tarefa de REN.

Souza, Nogueira e Lotufo (2019) empregaram uma arquitetura baseada na combinação do modelo *Bidirectional Encoder Representation from Transformers* (BERT) para a realização da tarefa de REN para português utilizando técnicas de treinamento baseadas em características e ajustes finos. Visando aproveitar a vantagem de poder ter contextos longos ao computar as representações de *tokens* usando o BERT, os autores utilizaram o contexto do documento inteiro e não somente da sequência de palavras. No treinamento o BERT foi usado nos dois tamanhos: *Base* e *Large*. Também foram utilizados os *corpora* brWaC, que possui 2,68 bilhões de *tokens* obtidos de 3,53 milhões de documentos, o HAREM I e o MiniHAREM nos cenários Total e Seletivo. Para o treinamento foram produzidos 3 modelos: *Multilingual BERT-Base*, *Portuguese BERT-Base* e *Portuguese BERT-Large*, sendo que estes foram avaliados e comparados com as arquiteturas CharWNN, LSTM-CRF e BiLSTM-CRF+FlairBBP. O modelo *Portuguese BERT-Large* com CRF obteve os melhores resultados para a Medida-F nos cenários Total e Seletivo.

Castro, Silva e Soares (2019) propuseram um sistema composto por uma rede do tipo *Bidirectional Long Short-Term Memory* (BiLSTM); *Conditional Random Fields* (CRF) e o ELMo (*Embeddings from Language Model*) – que foi utilizado para o domínio geral (extraído de textos da Wikipédia) e jurídico (retirado de documentos públicos da Justiça do Trabalho do Brasil) do idioma português – combinado com representações no nível de caracteres obtidas de uma *Convolutional Neural Network* (CNN). Os *corpora* usados foram: WikiNER, LeNER-Br, HAREM I, MiniHAREM, Paramopama e um *corpus* contendo documentos da Justiça do Trabalho do Brasil. Foram feitas algumas combinações – que foram chamadas de 2xELMo+CNN, ELMo+CNN e ELMo+CNN+Wang2Vec – e utilizadas para o treinamento e testes nos diversos *corpora* usados no trabalho. Após os testes, os resultados foram agrupados por modelo e a melhor Medida-F foi obtida pela combinação 2xELMo+CNN.

Silva (2020) explora a combinação entre traços linguísticos (características ortográficas, morfossintáticas e lexicais das palavras) e representações usando *Word Embeddings* em conjunto com uma rede neural BiLSTM com o objetivo de construir um modelo de classificação semântica para REN baseado na língua portuguesa. Adicionalmente foi feita uma análise da distribuição contextual de cada categoria de entidade no *corpus* HAREM. A autora adotou o BERT como modelo para representação das palavras, pois ele codifica a representação da palavra com base no contexto à direita e à esquerda dela. Dois modelos foram testados: o BERT na abordagem com ajuste fino (*fine-tuning*) e uma BiLSTM com *Word Embeddings* extraídos do BERT combinados a traços linguísticos

anotados manualmente. Após os experimentos, considerando o cenário Total e Seletivo do HAREM, a autora comparou os resultados com outros sistemas (CharWNN, BiLSTM-CNN, BiLSTM-CRF, BiLSTM-CRF+FlairBBP e BERT$_{PT-LARGE}$-CRF) e constatou que: assim como o sistema BERT$_{PT-LARGE}$-CRF obteve os melhores resultados entre os sistemas usados na comparação, o modelo baseado no BERT-PT apresentou desempenho melhor que as outras variações dos modelos desenvolvidos no trabalho.

Albuquerque et al. (2022) apresentaram dois *corpora* produzidos com base em documentos legislativos do tipo Projetos de Lei (PL) e Solicitações de Trabalho (ST) providos pela Câmara dos Deputados, que receberam os nomes UlyssesNer-Br_PL-corpus (teve o nome simplificado para PL-corpus) e UlyssesNer-Br_ST-corpus (simplificado para ST-corpus), respectivamente. Esses *corpora* foram anotados em três fases e com base em 5 categorias do HAREM (Pessoa, Localização, Organização, Evento e Data) e outras duas categorias definidas pelos autores que foram Fundamento da Lei e Produto da Lei. Experimentos foram realizados com os *corpora* criados utilizando os algoritmos de *Hidden Markov Model* (HMM) e CRF, e os resultados obtidos demonstraram que o CRF obteve melhores resultados do que o HMM.

2.3 MEDIDAS E MÉTODOS DE AVALIAÇÃO DE SISTEMAS DE REN

Ao longo dos anos, diversas conferências foram realizadas com o objetivo de promover competições para avaliação conjunta dos sistemas de REN com o intuito de fomentar pesquisas na área e avaliar o seu progresso (PIROVANI, 2019). Fazem parte da lista de conferências destacadas por Pirovani (2019): **(i)** *Message Understanding Conference* (MUC-6) ocorrida em 1995 onde a tarefa de REN foi formalizada; **(ii)** MUC-7 que introduziu em 1998 a tarefa de identificação de relações com organizações (*employee_of, product_of, location_of*). **(iii)** *Automatic Content Extraction* (ACE) que aconteceu entre os anos de 2000 e 2008, e explorou a extração de entidades, relações e eventos utilizando bases de dados multimídia (textos, áudios e imagens); **(iv)** *Conference on Natural Language Learning* (CoNLL) que em 2002 avaliou os idiomas espanhol e holandês, e em 2003 o inglês e o alemão; e **(v)** HAREM que foi inspirado na MUC, porém utilizando o idioma português, e aconteceu em duas edições: primeiro HAREM (entre 2004 e 2006) e segundo HAREM (entre 2007 e 2008).

Para avaliar a correspondência de detecção das entidades pode-se utilizar um *corpus* com as entidades de interesse anotadas, sendo denominado Coleção Dourada (CD). É possível avaliar um sistema de REN de duas formas: exata e flexível. Na forma exata todas as palavras que formam as entidades avaliadas precisam ser iguais às da CD, por outro lado, na forma flexível uma entidade pode ser considerada correta mesmo que alguns termos divirjam daqueles da entidade de referência constante na CD. Uma vez decidido o critério de correspondência (exata ou flexível), o desempenho de um sistema de REN

pode ser mensurado quantitativamente utilizando as tradicionais medidas de Cobertura (C), Precisão (P) e Medida-F (CASTRO, 2019). Essas medidas são computadas, conforme apresentadas nas Equações 1, 2 e 3:

$$C = \frac{VP}{VP + FN} \tag{1}$$

$$P = \frac{VP}{VP + FP} \tag{2}$$

$$\textit{Medida-F} = \frac{2 \times P \times C}{P + C} \tag{3}$$

Onde:

- Verdadeiro Positivo (VP) é a quantidade de entidades identificadas e classificadas corretamente;
- Falso Positivo (FP) é o total de entidades identificadas e classificadas de forma errada; e
- Falso Negativo (FN) é o total de entidades presentes no documento que não foram identificadas pelo modelo.

Segundo Castro (2019), apesar da Medida-F ser relevante para avaliar sistemas de REN, pois considera tanto a medida de Cobertura quanto a de Precisão, quando se trabalha com várias classes de entidades é importante considerar as diferentes ponderações da Medida-F. A média ponderada pode ser feita nos modos micro e macro. No modo micro a Precisão e Cobertura são calculadas de uma única vez e todos os valores de cada uma das classes são considerados; já no macro, calcula-se a Precisão e Cobertura de cada classe separadamente e só depois a média dos valores dos resultados é calculada. Essa distinção no modo de computar resulta em uma diferença entre os dois modos em que no micro as contribuições de desempenho das entidades têm o mesmo peso, mas no macro as contribuições de desempenho das classes é que têm o mesmo peso, independentemente do total de entidades pertencentes a cada classe.

Se o *corpus* usado para treinar o modelo de REN não estiver balanceado em relação à quantidade das entidades entre as classes de interesse, é preferível utilizar o modo micro,

pois o macro não considera este desequilíbrio e atribui o mesmo peso para todas as classes (CASTRO, 2019). Esses modos são calculados conforme apresentado nas Equações 4, 5, 6, 7, 8 e 9:

$$MicroC = \frac{\sum_{i=1}^{n} VP_i}{\sum_{i=1}^{n} VP_i + FN_i} \tag{4}$$

$$MicroP = \frac{\sum_{i=1}^{n} VP_i}{\sum_{i=1}^{n} VP_i + FP_i} \tag{5}$$

$$MicroF = \frac{2 \times MicroP \times MicroC}{MicroP + MicroC} \tag{6}$$

$$MacroC = \frac{\sum_{i=1}^{n} C_i}{n} \tag{7}$$

$$MacroP = \frac{\sum_{i=1}^{n} P_i}{n} \tag{8}$$

$$MacroF = \frac{2 \times MacroP \times MacroC}{MacroP + MacroC} \tag{9}$$

Onde,

- n é o número de classes de entidades avaliadas.

3 RECONHECIMENTO DE ENTIDADES NOMEADAS USANDO COMBINAÇÃO DE *EMBEDDINGS*

Conforme apresentado no Capítulo 2, arquiteturas baseadas em Aprendizado Profundo (AP), do inglês *Deep Learning* (DL), vêm sendo empregadas com sucesso em diversas aplicações de PLN, incluindo a tarefa de REN (LI et al., 2020) (YADAV; BETHARD, 2018). Modelos de DL são baseados em redes neurais e não requerem que atributos sejam gerados manualmente. Além disso, sistemas de REN usando redes neurais têm adotado representações de múltiplas dimensões (*embeddings*) no nível de palavras ou de caracteres para criar representações mais ricas (SANTOS et al., 2019). Esses modelos de *embeddings* são treinados para gerar representações tanto estaticamente, por exemplo, o modelo *Word2Vec* (MIKOLOV et al., 2013), quanto contextualmente usando Modelos de Linguagem (ML) baseados em redes neurais, como o modelo BERT (DEVLIN et al., 2019).

Os sistemas de REN mais recentes têm aplicado *embeddings* contextuais — obtidos de modelos de linguagem que utilizam redes neurais — com arquiteturas de DL, especialmente usando a rede do tipo *Long Short-Term Memory* (LSTM) combinada com o algoritmo CRF (SANTOS et al., 2019; SOUZA; NOGUEIRA; LOTUFO, 2019; LI et al., 2020). Os *embeddings* contextuais diferem dos tradicionais (estáticos), pois a cada palavra é atribuída uma representação que considera a sentença em que está incluída. Dessa forma, é possível capturar características semânticas e sintáticas do uso das palavras e como elas variam dependendo do contexto em que a palavra está sendo usada (polissemia) (PETERS et al., 2018).

Neste capítulo, é apresentada uma investigação da eficiência da combinação de *embeddings* tradicionais e contextuais para a tarefa de REN em textos escritos em português. Para isso, adotou-se uma arquitetura baseada em uma *Bidirectional Long Short-Term Memory* com o modelo CRF, chamada de $BiLSTM - CRF$. Foram investigadas a combinação de dois modelos tradicionais de *word embeddings*, Glove (PENNINGTON; SOCHER; MANNING, 2014) e Word2Vec (MIKOLOV et al., 2013), com os modelos contextualizados extraídos do FLAIR (AKBIK; BLYTHE; VOLLGRAF, 2018), BERT (DEVLIN et al., 2019) e ELMO (PETERS et al., 2018), todos pré-treinados para português. Os experimentos foram realizados utilizando três *corpora* disponíveis na literatura: HAREM (SANTOS et al., 2006), Paramopama (JÚNIOR et al., 2015) e LeNER-BR (ARAUJO et al., 2018). Diferentes cenários foram avaliados, desde o uso de um modelo com um único *word embedding* até a combinação dos *embeddings* tradicionais e contextualizados. A implementação e os dados usados nos experimentos estão disponíveis em um repositório do GitHub[6].

[6] <https://github.com/messias077/ner_pt>

3.1 *WORD EMBEDDINGS* ANALISADOS

O uso de *word embeddings* pré-treinados é um dos componentes fundamentais de uma abordagem de REN baseada em redes neurais profundas (LI et al., 2020). Neste trabalho, foram avaliados os modelos do Word2vec (MIKOLOV et al., 2013) e Glove (PENNINGTON; SOCHER; MANNING, 2014), brevemente descritos a seguir.

Word2vec (MIKOLOV et al., 2013). É uma técnica estatística usada para aprender representações distribuídas de palavras em textos. Ela é baseada em redes neurais e possui dois modelos: *Continuous Bag Of Words* (CBOW) e *Skip-gram*. O modelo CBOW é treinado para predizer uma palavra alvo p_t a partir de uma janela de palavras de contexto $p_{t-2}, p_{t-1}, p_{t+1}, p_{t+2}$. Enquanto isso, o modelo *Skip-gram* visa predizer palavras de contexto $p_{t-2}, p_{t-1}, p_{t+1}, p_{t+2}$ dada uma palavra alvo p_t. Ambos os modelos visam aprender uma representação n-dimensional de palavras considerando o contexto em que estão inseridas. O Word2Vec produz vetores de representações das palavras com base no *corpus* utilizado para o seu treinamento.

Glove (PENNINGTON; SOCHER; MANNING, 2014). É um modelo usado para prover representações de palavras distribuídas criando uma matriz quadrada de coocorrências de palavras e computando o número de vezes que cada palavra ocorre no mesmo contexto. O modelo visa construir uma representação distribuída das palavras de modo que as características aprendidas reflitam a relevância entre as palavras conforme as taxas de probabilidade condicional de ocorrências entre elas. A matriz de coocorrência das palavras utilizada para treinar o modelo é esparsa (possui grande quantidade de elementos com valor zero), por isso o modelo é treinado utilizando somente os elementos que não são zerados. Isso possibilita que o modelo seja treinado em um *corpus* grande com bilhões de palavras sem perda no desempenho.

Neste trabalho, foram adotados os modelos pré-treinados para o português do Brasil do Word2vec e Glove, disponibilizados por (HARTMANN et al., 2017) em[7]. Uma importante limitação de ambos os modelos é que eles geram vetores estáticos para representar as palavras. Assim, fenômenos como a polissemia não são considerados. Portanto, a aplicação de *embeddings* contextuais gerados a partir de modelos de linguagem tem crescido nos últimos anos. Neste trabalho, foram avaliados os seguintes modelos.

BERT (DEVLIN et al., 2019). *Bidirectional Encoder Representation from Transformers* (BERT) é um modelo de linguagem neural baseado na arquitetura *Transformer* (VASWANI et al., 2017). Esse modelo gera representações bidirecionais analisando os contextos dos lados esquerdo e direito das palavras. Ele possui duas arquiteturas básicas (*Base* e *Large*) que diferem em alguns parâmetros. A arquitetura *Base* compreende 12 camadas de

[7] <http://www.nilc.icmc.usp.br/embeddings>

codificação, 12 cabeças de atenção e 768 dimensões ocultas. Enquanto isso, a arquitetura *Large* possui 24 camadas de codificação, 16 cabeças de atenção e 1024 dimensões ocultas. O modelo BERT é treinado utilizando duas funções objetivos: **(i)** a modelagem de linguagem mascarada, do inglês *Masked-Language Modeling* (MLM); e **(ii)** a previsão da próxima frase, do inglês *Next Sentence Prediction* (NSP).

A MLM consiste em esconder aleatoriamente algumas palavras do texto e o objetivo é que o modelo aprenda a predizer as palavras escondidas utilizando como base o contexto onde a palavra está inserida. O uso da tarefa de MLM possibilita a junção dos contextos à esquerda e à direita das palavras, permitindo o pré-treino de um modelo bidirecional profundo. Na tarefa da NSP, durante o treinamento do BERT, as frases do documento original são embaralhadas e o modelo recebe pares de frases como entrada e aprende a prever se a segunda frase do par é a subsequente no documento original.

Neste trabalho, foram utilizados os modelos pré-treinados e disponibilizados por (SOUZA; NOGUEIRA; LOTUFO, 2020) em[8].

Elmo (PETERS et al., 2018). *Embeddings from Language Models* (ELMO) geram representações contextuais profundas a partir de um Modelo de Linguagem Bidirecional (biLM) baseado em uma rede BiLSTM. A entrada do modelo é uma representação em nível de caractere extraída por uma rede neural convolucional. O Elmo pode ser integrado facilmente a outros modelos, e ele difere de outros *word embeddings* porque cada palavra recebe uma representação baseada na sentença inteira a qual a palavra está inserida. As representações do Elmo são profundas, uma vez que são uma função de todas as camadas internas da rede bidirecional. Essas representações são ricas e podem capturar aspectos dependentes do contexto, do significado das palavras (podem ser utilizadas, sem modificação, para tarefas de desambiguação entre palavras) e aspectos de sintaxe (podem ser utilizadas para tarefas como a *POS-tagging*). Neste trabalho, foi adotado o modelo treinado por (CASTRO; SILVA; SOARES, 2019) para o idioma português e disponibilizado em[9].

Flair (AKBIK; BLYTHE; VOLLGRAF, 2018). O Flair *embedding* é um modelo em nível de palavra que lida com sequências de palavras e caracteres. Assim, esse modelo pode analisar o contexto de palavras vizinhas analisando os caracteres dessas palavras. Os *embeddings* são extraídos a partir de um modelo de linguagem no nível de caractere, do inglês *Character-level Language Modeling* (CharLM), baseado em redes neurais. O Flair combina dois modelos, um treinado para analisar a sequência de caracteres da esquerda para a direita (*forward*) e outro treinado na direção oposta, ou seja, da direita para a esquerda (*backward*). Por ser contextualizado, o Flair pode gerar representações diferentes para uma

[8] <https://github.com/neuralmind-ai/portuguese-bert>
[9] <https://allennlp.org/elmo>

mesma palavra caso a mesma seja empregada em contextos diferentes. As representações extraídos do Flair possuem as seguintes características: **(i)** são pré-treinadas em *corpora* grandes e não etiquetados; **(ii)** capturam o significado das palavras dependendo do contexto em que as mesmas forem empregadas; e **(iii)** modelam as palavras e os contextos como uma sequência de caracteres com o intuito de tratar palavras raras e com erros ortográficos. Neste trabalho, foram utilizados os modelos treinados por (SANTOS et al., 2019) e disponibilizados em[10] para português.

No Quadro 1 é apresentado um comparativo entre os *word embeddings* analisados que considera três características: **(i) Tipo**. Se utiliza vetores estáticos ou contextualizados; **(ii) Nível de Representação**. Indica se o *word embedding* representa os vetores considerando a palavra inteira ou cada um dos caracteres da palavra; e **(iii) Considera Polissemia**. Informa se o modelo consegue diferenciar os vários significados de uma palavra, caso exista mais de um.

Quadro 1 – Comparativo entre os *word embeddings* analisados.

Word Embedding	**Característica**		
	Tipo	**Nível de Representação**	**Considera Polissemia**
Word2vec	Estático	Palavra	Não
Glove	Estático	Palavra	Não
BERT	Contextual	Palavra	Sim
Elmo	Contextual	Caractere	Sim
Flair	Contextual	Palavra e Caractere	Sim

Fonte: Elaborado pelo autor (2022).

3.2 ARQUITETURA DO MODELO DE CLASSIFICAÇÃO ADOTADO

Nesta seção é descrita a arquitetura do modelo de classificação de entidades nomeadas utilizada na execução dos experimentos. Inicialmente, na Subseção 3.2.1 são detalhados os componentes empregados para construção de uma arquitetura do tipo BiLSTM-CRF. Posteriormente, na Subseção 3.2.2 é descrito o *framework* FLAIR, que utiliza essa arquitetura.

3.2.1 BiLSTM-CRF

Hochreiter e Schmidhuber (1997) introduziram uma nova arquitetura de rede neural recorrente (tipo de rede capaz de aprender e armazenar as dependências entre dados sequenciais e temporais), chamada de *Long Short-Term Memory* (LSTM). O objetivo da rede LSTM era resolver algumas limitações que os algoritmos empregados nas redes neurais recorrentes anteriores não haviam resolvido. Duas dessas limitações eram: **(i)**

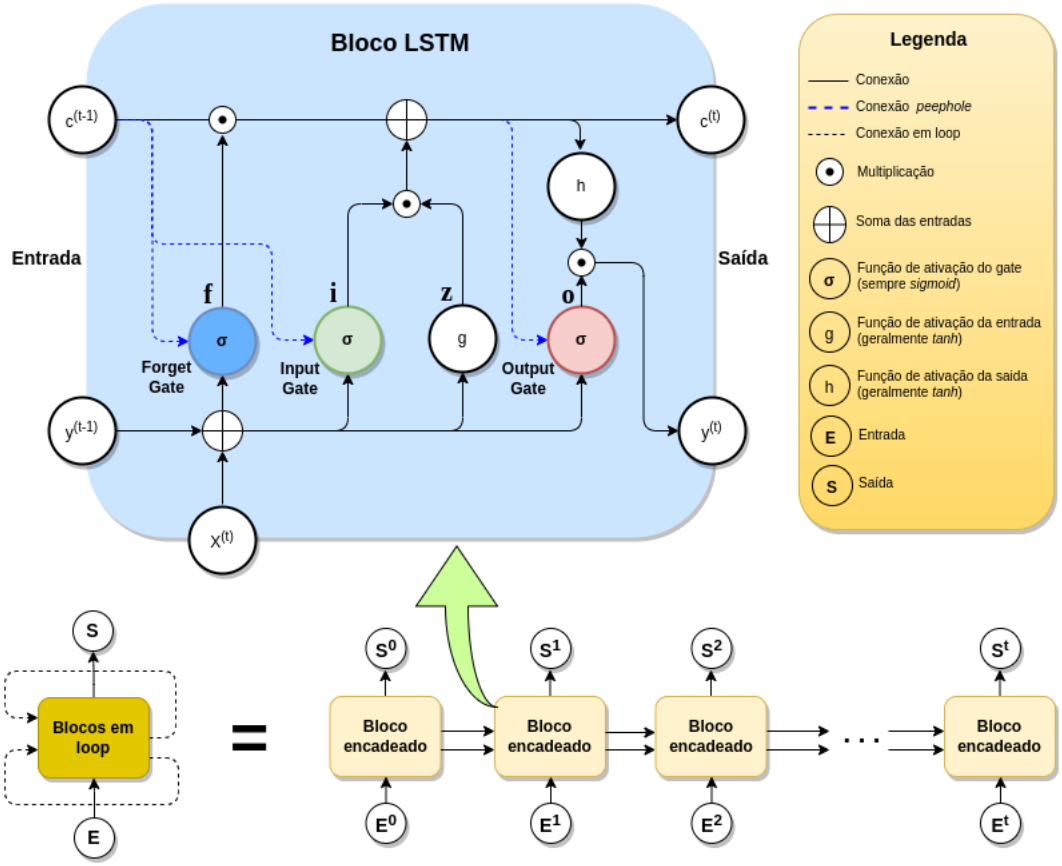
[10] <https://github.com/jneto04/ner-pt>

explosão dos valores dos erros retropropagados, os valores dos erros aumentavam muito, podendo causar oscilações nos pesos; e **(ii)** dissipação dos erros retropropagados, onde os valores dos erros diminuíam tanto que faziam com que a rede levasse muito tempo no processo de treinamento ou até mesmo não funcionasse. Em ambos casos, os valores dos erros dependem do tamanho dos pesos das redes. Para resolver esses problemas, a rede LSTM emprega um algoritmo que impõe um fluxo de erro constante (evitando a explosão e a dissipação dos erros) com o apoio dos componentes: célula de memória e *gate units*, que compreendem portas de entrada (*input gates*) e portas de saída (*output gates*).

As redes do tipo LSTM são utilizadas em diversos domínios de aplicações, tais como previsão de séries temporais, reconhecimento de texto, processamento de linguagem natural, visão computacional, inclusão de legendas em imagens e vídeos, dentre outras aplicações (LI et al., 2020). Uma rede do tipo LSTM padrão é composta por células de memória (responsáveis por manter o seu próprio estado ao logo do tempo e regular o fluxo de informações com ajuda dos *gate units*), *input gates*, *output gates* e *forget gates*. O *forget gate* não fazia parte da arquitetura inicial, proposto por Gers, Schmidhuber e Cummins (2000) para permitir que a rede pudesse esquecer gradualmente informações que não estavam sendo mais atualizadas, portanto, não eram mais úteis. Gers e Schmidhuber (2000) também propuseram a inclusão de "*peephole connections*", conexões que permitem aos *gates units* acessarem o estado das células (HOUDT; MOSQUERA; NÁPOLES, 2020).

Os componentes presentes em uma LSTM padrão são apresentados na Figura 3. Percebe-se, nessa figura, que a rede é composta por blocos interligados. Em seguida, esses blocos foram dispostos sequencialmente na figura para ilustrar que a saída de um bloco hipotético B_i é interligada à entrada do bloco B_{i+1} cuja saída é ligada à entrada do bloco B_{i+2} e assim segue até que todos os blocos sejam conectados entre si. Na parte superior é destacada a estrutura de um bloco LSTM e seus componentes internos. As letras **f**, **i**, **z** e **o**, representam as funções de ativação dos *gate units* e a função de ativação da entrada.

Figura 3 – Arquitetura de uma rede LSTM padrão.



Fonte: Adaptado de Houdt, Mosquera e Nápoles (2020).

Houdt, Mosquera e Nápoles (2020) descrevem a computação de um passo de ida (*forward*) de uma rede LSTM hipotética, baseada na Figura 3, com N blocos de processamento e M entradas, conforme a seguir.

Entrada do bloco. Neste momento o componente de entrada do bloco é atualizado, onde são combinados a entrada corrente $x^{(t)}$ e a saída da última iteração que está chegando em $y^{(t-1)}$. Esse processo é feito conforme a Equação 10.

$$z^{(t)} = g(W_z x^{(t)} + R_z y^{(t-1)} + b_z) \tag{10}$$

Onde, W_z e R_z são os pesos associados ao $x^{(t)}$ e $y^{(t-1)}$, respectivamente, enquanto b_z representa o vetor de *bias*.

Input gate. Momento em que o *input gate* é atualizado através da combinação da entrada corrente $x^{(t)}$, a saída da última iteração que está chegando em $y^{(t-1)}$ e o valor da célula $c^{(t-1)}$ obtido na última iteração. Esse procedimento é computado utilizando a Equação 11.

$$i^{(t)} = \sigma(W_i x^{(t)} + R_i y^{(t-1)} + p_i \odot c^{(t-1)} + b_i) \tag{11}$$

Em que $\odot$ representa a multiplicação de dois vetores elemento a elemento, W_i, R_i e p_i são os pesos associados com $x^{(t)}$, $y^{(t-1)}$ e $c^{(t-1)}$, respectivamente, e b_i representa o vetor de *bias* associado a este componente.

Nas duas etapas anteriores, a camada LSTM determina quais informações serão retidas nos estados das células da rede $c^{(t)}$. Também foram selecionados os valores candidatos $z^{(t)}$ que poderiam, potencialmente, serem adicionados aos estados da célula e os valores de ativação $i^{(t)}$ dos *input gates*.

Forget gate. Nesta etapa, a unidade LSTM determina quais informações devem ser esquecidas, e por conta disso removidas dos estados das células anteriores $c^{(t-1)}$. Portanto, os valores de ativação $f^{(t)}$ do *forget gates* no passo de tempo t são calculados com base na entrada atual $x^{(t)}$, nas saídas $y^{(t-1)}$ e no estado $c^{(t-1)}$ das células de memória no passo de tempo anterior $(t-1)$, as conexões *peephole* e os *bias* b_f dos *forget gates*. Esse procedimento é computado conforme a Equação 12.

$$f^{(t)} = \sigma(W_f x^{(t)} + R_f y^{(t-1)} + p_f \odot c^{(t-1)} + b_f) \tag{12}$$

Onde, W_f, R_f e p_f são os pesos associados com $x^{(t)}$, $y^{(t-1)}$ e $c^{(t-1)}$, respectivamente, e b_f é o vetor de *bias*.

Célula. Aqui é computado o valor da célula, onde são combinados os valores da entrada do bloco $z^{(t)}$, do *input gate* $i^{(t)}$ e do *forget gate* $f^{(t)}$, juntamente com o valor da célula anterior. Esse procedimento é calculado conforme a Equação 13.

$$c^{(t)} = z^{(t)} \odot i^{(t)} + c^{(t-1)} \odot f^{(t)} \tag{13}$$

Output gate. Neste momento o *output gate* é calculado, através da combinação da entrada corrente $x^{(t)}$, a saída da unidade LSTM $y^{(t-1)}$ e o valor da célula $c^{(t-1)}$ da iteração anterior, conforme a Equação 14.

$$o^{(t)} = \sigma(W_o x^{(t)} + R_o y^{(t-1)} + p_o \odot c^{(t-1)} + b_o) \tag{14}$$

Onde, W_o, R_o e p_o são os pesos associados com $x^{(t)}$, $y^{(t-1)}$ e $c^{(t-1)}$, respectivamente, e b_o representa o vetor de *bias*.

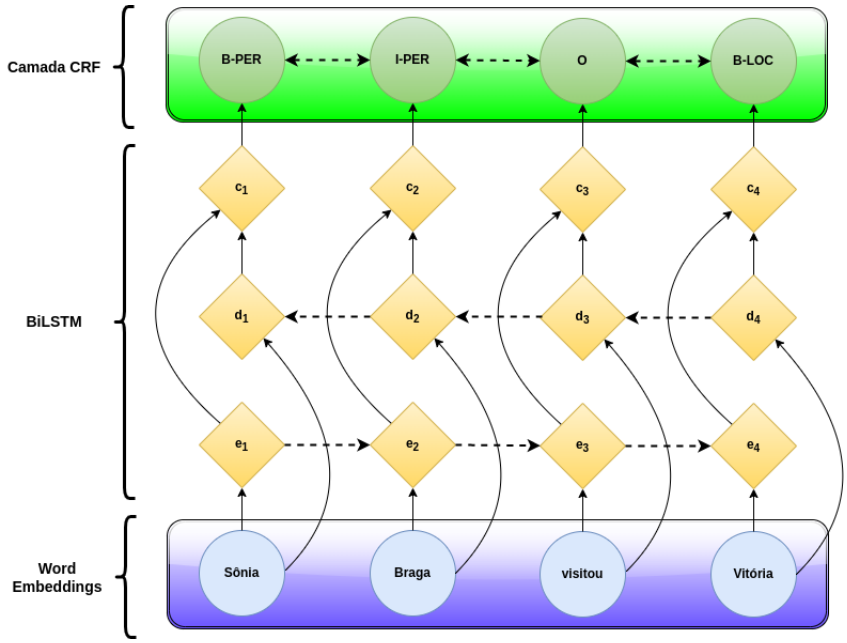
Saída do bloco. Por último, a saída do bloco é calculada, onde são combinados o valor corrente da célula $c^{(t)}$ com o valor corrente do *output gate* $o^{(t)}$, conforme apresentado na Equação 15.

$$y^{(t)} = h(c^{(t)}) \odot o^{(t)} \tag{15}$$

Nas etapas descritas anteriormente, σ, g e h denotam funções de ativação não-lineares. A função logística $\sigma(x) = \frac{1}{1+e^{1-x}}$ é utilizada como função de ativação dos *gate units*, já para a ativação das entradas e saídas dos blocos é geralmente usada a função tangente hiperbólica ($tanh(x)$) que nessas etapas é representada por $g(x)$ e $h(x)$.

Lample et al. (2016) construíram uma arquitetura baseada em uma rede LSTM bidirecional (BiLSTM) em conjunto com o algoritmo CRF, denominada de BiLSTM-CRF, aplicada na tarefa de REN. Essa arquitetura é apresentada na Figura 4 em que acontece a junção de duas redes LSTM, uma na direção para frente (*forward*) e a outra na direção para trás (*backward*), formando assim uma camada BiLSTM, que está posicionada no centro da figura. A última camada é formada pelo algoritmo CRF que recebe as saídas da rede BiLSTM. A primeira camada da arquitetura é composta pela representação multidimensional *word embeddings* que fornecem as entradas para a rede BiLSTM.

Figura 4 – Arquitetura BiLSTM-CRF.



Fonte: Adaptado de Lample et al. (2016).

Dada uma frase $(X_1, X_2, ..., X_n)$ com n palavras, onde cada uma delas é representada por um vetor d-dimensional, a rede LSTM computa a representação $\overrightarrow{h_t}$, com base no contexto de cada palavra t partindo do início da sequência para o final. Também é possível gerar uma representação considerando o contexto das palavras a partir do final da sequência em direção ao início dela ($\overleftarrow{h_t}$), podendo capturar mais informações úteis. Dessa forma, é possível obter uma representação mais rica das palavras ao utilizar-se de duas redes LSTM, uma para gerar a representação $\overrightarrow{h_t}$ e outra para gerar $\overleftarrow{h_t}$ e em seguida concatenar essas duas representações, $h_t = [\overrightarrow{h_t}; \overleftarrow{h_t}]$. Essa concatenação h_t consegue capturar informações sobre o contexto das palavras nas sequências, sendo útil para realizar tarefas de marcação das palavras (LAMPLE et al., 2016).

De acordo com Lample et al. (2016), a arquitetura mostrada na Figura 4 funciona da seguinte forma: **(i)** são gerados *word embeddings* para cada palavra nas sentenças, que nesse exemplo é a frase *"Sônia Braga visitou Vitória"*, e passados como entrada para uma rede BiLSTM; **(ii)** a rede BiLSTM gera as representações da esquerda (do início para o fim da sequência) e_i, no exemplo representado por e_1, e_2, e_3 e e_4, e da direita (do fim para o início da sequência) d_i, representado por d_1, d_2, d_3 e d_4; **(iii)** as representações e_i e d_i são concatenadas e formam as representações das palavras em um contexto c_i, que no exemplo dado é c_1, c_2, c_3 e c_4; **(iv)** as representações dos contextos c_i são passadas como entrada para a camada CRF que prediz os rótulos de cada palavra considerando os rótulos das palavras vizinhas, sendo que nesse exemplo a palavra *Sônia* recebeu o rótulo *B-PER*, que indica ser o início de uma entidade do tipo **Pessoa**; a palavra *Braga* recebeu o rótulo *I-PER*, indicando ser parte de uma entidade do tipo **Pessoa** composta por mais de uma palavra; *visitou* recebeu o rótulo *O*, que informa que esta palavra não é uma entidade; e *Vitória* teve o rótulo *B-LOC* atribuído-lhe, que indica ser uma entidade do tipo **Localização**.

3.2.2 Flair

FLAIR[11] é um *framework* de código aberto implementado por Akbik et al. (2019) com o objetivo de prover uma interface simples e unificada para utilizar diferentes tipos de *word embeddings* sem que seja preciso preocupar-se com os detalhes específicos da arquitetura desses *embeddings*, possibilitando a combinação de vários desses, sem muito esforço. Esse *framework* também traz outras facilidades como rotinas para escolha de hiperparâmetros, módulo para obter base de dados de PLN públicas para serem utilizados na ferramenta, modelos pré-treinados, entre outras.

O FLAIR foi utilizado para realizar a classificação das entidades nomeadas porque implementa um modelo de rotulagem de sequências usando uma BiLSTM-CRF, que tem sido usado com sucesso em vários trabalhos anteriores para a tarefa REN (CASTRO;

[11] <https://github.com/flairNLP/flair>

SILVA; SOARES, 2019; SANTOS et al., 2019). Além disso, o *framework* FLAIR permite a concatenação de diferentes tipos de *embeddings* para criar uma representação combinada. Essa estratégia, denominada *Stacking Embeddings*, tem se mostrado eficaz e tem apresentado resultados promissores em trabalhos anteriores (AKBIK; BLYTHE; VOLLGRAF, 2018; SANTOS et al., 2019). Assim, a combinação dos *embeddings* é dada pela Equação 16.

$$w_{comb} = \begin{bmatrix} w^T \\ \\ w^C \end{bmatrix} \tag{16}$$

Sendo que, w^T é extraído de *embeddings* tradicionais (neste trabalho do Word2vec ou Glove), e w^C é obtido de *embeddings* contextualizados (neste trabalho a partir do BERT, ELMO ou Flair). A representação de *embeddings* combinados w_{comb} é passada como entrada para o modelo BiLSTM-CRF para realizar a identificação das entidades. O modelo CRF fornece então a probabilidade final da sequência sobre os rótulos possíveis.

3.3 CORPORA

Nesta seção são apresentados os *corpora* utilizados nos experimentos realizados neste trabalho. Esses *corpora* foram adotados por estarem disponíveis na literatura e por serem amplamente usados em trabalhos de REN para o português (SOUZA; NOGUEIRA; LOTUFO, 2019; SANTOS et al., 2019).

Harem (SANTOS et al., 2006). Um dos principais *corpora* utilizados para a avaliação de sistemas REN em português é a coleção do HAREM. A primeira versão do HAREM foi dividida em dois subconjuntos conhecidos como Primeiro HAREM e MiniHAREM. Cada conjunto foi anotado manualmente e teve dois cenários de avaliação. O primeiro cenário, denominado Total, possui anotações para dez categorias de entidades: Local, Pessoa, Organização, Valor, Tempo, Obra, Coisa, Acontecimento, Abstração e Outros. O segundo cenário, conhecido como Seletivo, possui anotações de apenas cinco classes: Pessoa, Organização, Local, Valor e Tempo.

Paramopama (JÚNIOR et al., 2015). Esse *corpus* foi criado a partir da extensão de outra base de dados denominada WikiNER, que possui versões em vários idiomas, inclusive para o português. Primeiro, os autores realizaram uma revisão e reclassificação das anotações incorretas do WikiNER português. Adicionalmente, o tamanho do *corpus* foi ampliado com a incorporação de textos obtidos em sites de notícias sobre economia, esportes, política, tecnologia, entre outros assuntos. Os textos foram anotados em quatro

categorias: Pessoa, Local, Organização e Tempo. Os autores realizaram vários experimentos combinando o *corpus* Paramopama com diferentes versões dos *corpora* HAREM.

LeNER-BR (ARAUJO et al., 2018). É um *corpus* de domínio específico construído a partir de 70 documentos legais, incluindo 66 documentos de diversos tribunais brasileiros e quatro documentos legislativos. Os documentos foram anotados manualmente, considerando as seguintes categorias de entidades: Organização, Pessoa, Tempo, Local, Legislação e Jurisprudência.

Assim como em outros trabalhos (CASTRO; SILVA; SOARES, 2018; SOUZA; NOGUEIRA; LOTUFO, 2019; SANTOS et al., 2019), o primeiro HAREM foi usado como conjunto de treinamento e o MiniHAREM como conjunto de teste. Foi adotada a versão pré-processada do *corpus* HAREM disponibilizado por (SOUZA; NOGUEIRA; LOTUFO, 2019) em[12]. O *corpus* Paramopama combinado com a segunda versão do HAREM disponibilizado em[13] também foi utilizado. Nesse *corpus*, as sentenças foram divididas mantendo a proporção entre as categorias de entidades, sendo 60% para treinamento, 20% para validação e 20% para teste. Além disso, as frases duplicadas foram removidas. Por fim, as sentenças duplicadas entre os conjuntos de treinamento, teste e validação identificadas no *corpus* LeNER-BR foram removidas. Na Tabela 2 são apresentadas algumas estatísticas descritivas das versões utilizadas desses *corpora*.

Tabela 2 – Estatísticas descritivas dos *corpora* adotados.

Corpus	**Conjunto**	**Sentenças**	**Palavras**	**Entidades**
HAREM (Total)	Treino	4.505	93.627	1.402
	Validação	237	4.685	113
	Teste	3.393	66.547	1.111
HAREM (Seletivo)	Treino	4.506	93.470	1.182
	Validação	238	4.830	94
	Teste	3.393	66.572	945
Paramopama + Segundo Harem	Treino	9.374	230.497	4.112
	Validação	3.125	76.235	1.752
	Teste	3.125	76.428	1.698
LeNER-BR	Treino	5.897	203.379	1.771
	Validação	888	34.636	303
	Teste	1.169	43.727	358

Fonte: Elaborado pelo autor (2022).

3.4 EXPERIMENTOS

Experimentos foram realizados para: **(i)** avaliar a estratégia de combinar *embeddings* tradicionais e contextuais; e **(ii)** comparar os resultados obtidos pela melhor configuração

[12] <https://github.com/jneto04/ner-pt>
[13] <https://github.com/davidsbatista/NER-datasets/tree/master/Portuguese>

encontrada com alguns sistemas de REN que alcançaram o estado da arte para o português. Antes de discutir os resultados obtidos, uma breve descrição do ambiente experimental e detalhes de implementação são apresentados na subseção a seguir.

3.4.1 Configuração dos Experimentos

No Quadro 2 são apresentados os hiperparâmetros adotados nos experimentos. Durante o treinamento foi utilizado o otimizador *Stochastic Gradient Descent With Weight Decay* (SGDW) e também foi definido um módulo para monitorar e persistir apenas o modelo com melhor desempenho (perda mínima) com base no conjunto de validação. Além disso, devido à estratégia de decaimento da taxa de aprendizado, o treinamento é interrompido se o valor da taxa de aprendizado for inferior a $0,002$. O tamanho do lote (*batch size*) foi definido com o valor 32 para quase todas as configurações, com exceção das que utilizaram os modelos BERT (*base* e *large*), cujo valor foi 16.

Quadro 2 – Hiperparâmetros adotados nos experimentos.

Hiperparâmetros	**Valor**
Taxa de Aprendizado Inicial	0,1
Otimizador	SGDW
Camadas Ocultas	256
Tamanho do Lote	32 ou 16
Quantidade Máxima de Épocas	100

Fonte: Elaborado pelo autor (2022).

Medidas de avaliação. Todas as medidas de avaliação usadas nos experimentos foram calculadas usando o *script* de avaliação disponibilizado na conferência CoNLL 2002[14]. Foram adotadas as medidas de avaliação micro (Precisão, Cobertura, e Medida-F) em nível de entidade, considerando apenas correspondências exatas. Essas medidas foram escolhidas por serem comumente usadas na literatura para avaliação de sistemas de REN (SOUZA; NOGUEIRA; LOTUFO, 2019; SANTOS et al., 2019).

Os experimentos foram executados em um computador com processador Intel Core i9, 32 Gigabytes (GB) de memória RAM, GPU Nvidia GeForce RTX 2060 e rodando o Ubuntu 20.04.

3.4.2 Resultados dos Experimentos

Neste primeiro experimento, foram avaliados diferentes cenários de aplicação (simples e combinados) de modelos de *embedding* tradicionais (Glove e Word2vec Skip-gram) e contextuais (BERT *base*, BERT *Large*, ELMO e Flair). Na Tabela 3 são apresentados os resultados obtidos com base na Medida-F, considerando os cenários de avaliação para cada

[14] <https://www.clips.uantwerpen.be/conll2002/ner/>

um dos *corpora* adotados: HAREM (Total e Seletivo), Paramopama + Segundo HAREM (referido apenas como Paramopama para simplificar) e LeNER-BR.

Tabela 3 – Resultados dos experimentos (%) das combinações dos *embeddings* avaliados com base na Medida-F. O maior resultado em cada *corpus* está destacado em negrito.

Tradicional	**Contextualizado**	**Harem Total**	**Harem Seletivo**	**Paramopama**	**LeNER-BR**
-	$BERT_{Base}$	75,95	82,96	87,87	91,08
	$BERT_{Large}$	77,20	82,58	88,07	91,95
	ELMO	73,62	80,17	88,99	88,61
	FLAIR	73,48	79,74	86,74	89,52
Glove	**-**	60,45	64,97	82,75	84,89
	$BERT_{Base}$	78,02	81,31	88,51	90,53
	$BERT_{Large}$	77,76	82,89	88,53	92,45
	ELMO	74,81	80,40	88,71	88,57
	FLAIR	72,91	78,85	88,22	90,93
Word2vec Skip	**-**	62,34	66,69	82,26	83,98
	$BERT_{Base}$	78,09	83,32	88,29	92,11
	$BERT_{Large}$	**78,40**	**83,88**	88,08	**92,71**
	ELMO	75,26	79,96	**89,14**	89,39
	FLAIR	74,29	80,43	88,18	90,68

Fonte: Elaborado pelo autor (2022).

Em relação aos cenários utilizando apenas os *embeddings* tradicionais, é possível observar que o *Word2vec Skip-gram* obteve os melhores resultados no *corpus* HAREM nos cenários Total e Seletivo, enquanto o Glove apresentou o melhor desempenho no Paramopama e no LeNER-BR. Considerando a adoção apenas dos *embeddings* contextuais, os modelos $BERT_{Base}$ e $BERT_{Large}$ obtiveram o melhor desempenho no *corpus* Seletivo e Total do HAREM, respectivamente. O $BERT_{Large}$ também obteve o melhor desempenho no LeNER-BR e o ELMO apresentou o melhor resultado no Paramopama. No geral, os *embeddings* contextuais apresentaram resultados superiores aos tradicionais em todos os *corpora* avaliados.

Analisando os 16 cenários avaliados, considerando apenas os *embeddings* contextuais (4 *corpora* $\times$ 4 modelos de *embeddings*), é possível observar que a combinação com o Glove levou a melhores resultados em 10 dos 16 cenários (62,5%). Enquanto isso, a integração com o *Word2vec Skip-gram* melhorou em 15 dos 16 cenários (93,75%). No geral, a arquitetura usando o $BERT_{Large}$ combinado com o *Word2vec Skip-gram* atingiu o melhor desempenho em ambos cenários do HAREM e no LeNER-BR, enquanto a combinação do ELMO com o *Word2vec Skip-gram* obteve o melhor desempenho no Paramopama.

Com base nos resultados experimentais obtidos, conclui-se que: **(i)** a combinação de *embeddings* tradicionais e contextuais levou a melhores resultados na maioria dos cenários

avaliados, especialmente usando *Word2vec Skip-gram*; e **(ii)** a arquitetura usando $BERT_{Large}$ com o *Word2vec Skip-gram* (BiLSTM-CRF $BERT_{Large}$-W2V) obteve o melhor desempenho geral considerando todos os *corpora* analisados.

Na Tabela 4 são apresentados os resultados detalhados por entidade com base nas medidas Cobertura, Precisão e Medida-F, considerando somente a arquitetura BiLSTM-CRF $BERT_{Large}$-W2V e os *corpora* Harem Total e Seletivo. É possível observar que os maiores valores obtidos pela arquitetura foram para a entidade Tempo (TMP). Observa-se também que os valores mais altos estão concentrados nas entidades que estão presentes simultaneamente em ambos os cenários. Uma possível razão para isso é que as menções das entidades dessas classes possuem um padrão mais simples que foi bem capturado pelo modelo.

Tabela 4 – Resultados dos experimentos da arquitetura que apresentou o melhor desempenho (BiLSTM-CRF $BERT_{Large}$-W2V) avaliados com base na Precisão (%), Cobertura (%) e Medida-F por tipo de entidade nos *corpora* Harem Total e Seletivo. Os maiores resultados em cada *corpus* estão destacados em negrito.

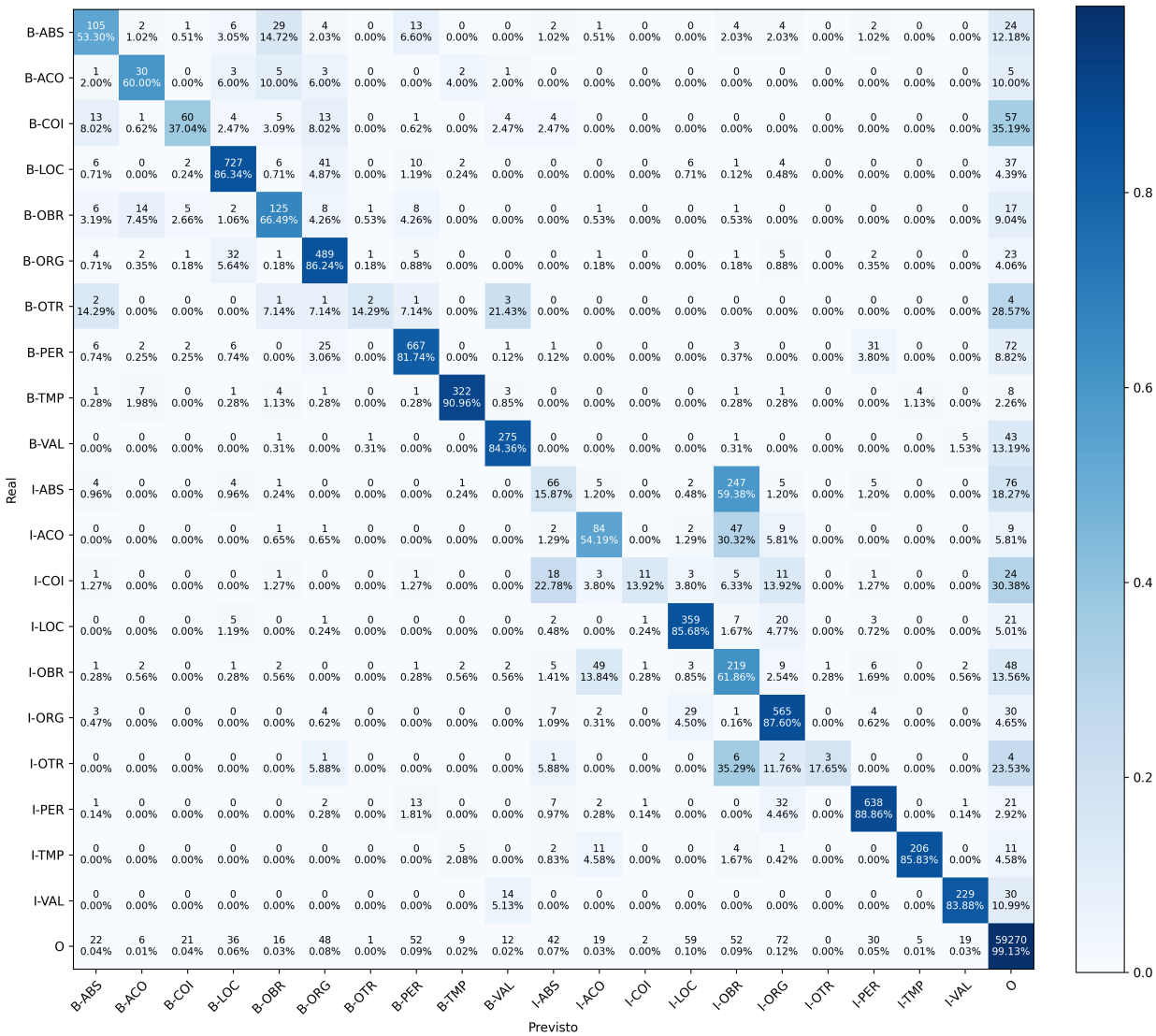
Entidades	**Harem Total**			**Harem Seletivo**		
	Precisão	**Cobertura**	**Medida-F**	**Precisão**	**Cobertura**	**Medida-F**
Abstração (ABS)	57,39	51,27	54,16	-	-	-
Acontecimento (ACO)	39,39	52,00	44,83	-	-	-
Coisa (COI)	64,13	36,42	46,46	-	-	-
Local (LOC)	86,82	85,27	86,04	86,20	84,56	85,37
Obra (OBR)	57,58	60,64	59,07	-	-	-
Organização (ORG)	73,68	83,42	78,25	77,08	80,07	78,55
Outros (OTR)	16,67	7,14	10,00	-	-	-
Pessoa (PER)	85,64	81,13	83,32	87,14	80,51	83,69
Tempo (TMP)	**92,71**	**89,83**	**91,25**	**93,20**	**88,98**	**91,04**
Valor (VAL)	80,63	77,91	79,25	84,03	80,67	82,32

Fonte: Elaborado pelo autor (2022).

Outro fato observado é que a arquitetura analisada não conseguiu um desempenho satisfatório em relação à entidade *Outros* (OTR), sendo que os valores de Precisão, Cobertura e Medida-F ficaram todos com valores menores do que 17%. Um dos motivos principais para esse baixo desempenho é a escassez de exemplos na base de dados de treinamento, que são de apenas 34 exemplos. Dentre as entidades que estão presentes somente no *corpus* Harem Total, o maior valor de Medida-F obtido pela arquitetura foi para a classe *Obra* (OBR) com 59,07%.

Para melhor entender os erros e acertos cometidos pelo modelo, é apresentada na Figura 5 a matriz de confusão gerada utilizando as etiquetas reais e previstas pela arquitetura BiLSTM-CRF $BERT_{Large}$-W2V com base no *corpus* Harem Total.

Figura 5 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Large}$-W2V no Harem Total.



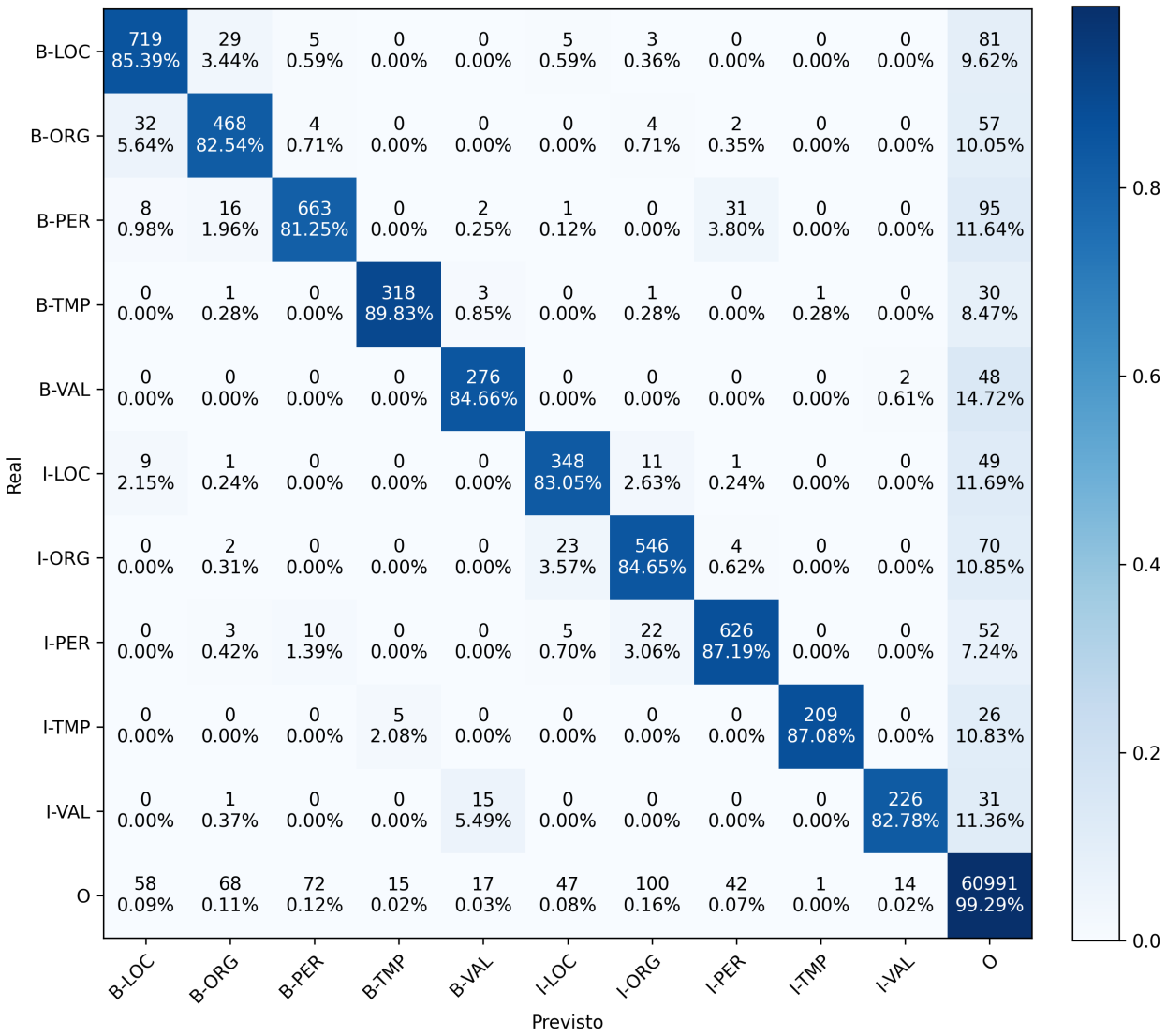
Fonte: Elaborado pelo autor (2022).

É possível observar na diagonal principal da matriz de confusão, que o modelo, em geral, estima corretamente os rótulos das entidades. Contudo, nota-se que houve certa dificuldade em distinguir algumas etiquetas, por exemplo, as acurácias obtidas nas classificações das etiquetas *B-OTR*, *I-OTR*, *I-ABS* e *I-COI* ficaram com valores menores do que 18%, onde a etiqueta *B-OTR* obteve as maiores taxas de confusão com O (28,57%) e *B-VAL* (21,43%); a etiqueta *I-OTR*, foi mais confundida com *I-OBR* (35,29%) e *O* (23,53%); para a etiqueta *I-ABS* as taxas maiores de confusão foram com *I-OBR* (59,38%) e *O* (18,27%); já a etiqueta *I-COI* obteve as maiores taxas de erro com *O* (30,38%) e *I-ABS* (22,78%).

Na Figura 6 é apresentada a matriz de confusão confeccionada usando a arquitetura BiLSTM-CRF $BERT_{Large}$-W2V, mas com base no *corpus* Harem Seletivo. Diferentemente ao que foi constatado no *corpus* Harem Total, nota-se que a diagonal dessa matriz está

melhor definida, porém deve-se ao fato de que as etiquetas que obtiveram uma alta taxa de confusão no Harem Total não estão presentes no cenário Seletivo. Mesmo assim, é possível observar que o modelo apresenta alguns erros, especialmente atribuindo o rótulo *O* para palavras que pertencem às entidades nomeadas.

Figura 6 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Large}$-W2V no Harem Seletivo.



Fonte: Elaborado pelo autor (2022).

Considerando somente as etiquetas constantes em ambos *corpora*, as acurácias desses rótulos na matriz de confusão apresentadas na Figura 5 foram próximas das respectivas etiquetas mostradas na matriz de confusão da Figura 6. A taxa de confusão das etiquetas que representam entidades com os rótulos que indicam que a palavra não é uma entidade (O) ficou em média 12,37% para a matriz de confusão do Harem Total e 10,65% para a matriz de confusão do Harem Seletivo.

Por fim, na Tabela 5 são apresentados os resultados baseados nas medidas de Precisão,

Cobertura e Medida-F (F1) da comparação entre a melhor arquitetura identificada neste trabalho (BiLSTM-CRF $BERT_{Large}$-W2V) com sistemas de REN que alcançaram resultados do estado da arte para o idioma português no *corpus* HAREM em ambos os cenários.

Tabela 5 – Comparação dos resultados (Precisão, Cobertura e F1 em %) com outros sistemas de REN da literatura. O maior resultado em cada medição e *corpus* está destacado em negrito.

Sistemas	**Harem Total**			**Harem Seletivo**		
	Precisão	**Cobertura**	**F1**	**Precisão**	**Cobertura**	**F1**
CRF	59,43	50,66	54,69	66,76	51,86	58,38
CharWNN(SANTOS; GUIMARÃES, 2015)	67,16	63,74	65,41	73,98	68,68	71,23
BiLSTM-CRF (CASTRO; SILVA; SOARES, 2018)	72,28	68,03	70,33	78,26	74,39	76,27
BiLSTM-CRF+FlairBBP (SANTOS et al., 2019)	74,91	74,37	74,64	83,38	81,17	82,26
PT-$BERT_{Large}$-CRF (SOUZA; NOGUEIRA; LOTUFO, 2019)	**80,08**	77,31	**78,67**	84,82	81,72	83,24
BiLSTM-CRF $BERT_{Large}$-W2V	79,29	**77,53**	78,40	**85,14**	**82,65**	**83,88**

Fonte: Elaborado pelo autor (2022).

O sistema PT-$BERT_{Large}$-CRF desenvolvido por Souza, Nogueira e Lotufo (2019) foi o que apresentou o melhor desempenho nas medidas de Precisão e F1 para o cenário Total do HAREM, enquanto que a arquitetura BiLSTM-CRF $BERT_{Large}$-W2V obteve o melhor resultado para a medida de Cobertura no HAREM (cenário Total) e todas as medidas considerando o cenário Seletivo. Assim, em geral, o BiLSTM-CRF $BERT_{Large}$-W2V superou os sistemas REN considerados estado da arte, principalmente no cenário Seletivo, e alcançou desempenho competitivo com o PT-$BERT_{Large}$-CRF no cenário Total.

4 RECONHECIMENTO DE ENTIDADES NOMEADAS EM DOCUMENTOS DE EDITAIS DE COMPRAS

Para o desenvolvimento deste projeto houve a necessidade da construção de alguns *corpora*, compostos por um conjunto de textos contendo as anotações das entidades de interesse persistidas para serem posteriormente utilizadas nas rotinas de treinamento e testes dos modelos de REN. A opção pela construção dos *corpora* foi motivada pelas particularidades encontradas nos editais utilizados para extração dos textos, como: **(i)** textos contendo regras a serem seguidas para entregas de bens e serviços, e **(ii)** arcabouço jurídico específico para formalizações de compras e contratações de serviços.

No artigo 40 da lei Federal nº 8.666/93, são definidas várias informações que um edital deve trazer em sua estrutura, por exemplo, objeto da licitação, prazo e condições para assinatura do contrato, sanções para o caso de inadimplemento, condições de pagamento, além de várias outras (BRASIL, 1993). Com base nessa estrutura definida em lei, e para fins específicos deste trabalho, as informações dos editais foram classificadas em dois grupos: **Jurídicas** e **Técnicas**.

As **informações jurídicas** apresentam os detalhes relacionados aos trâmites jurídicos e administrativos necessários para a condução da licitação, como leis, decretos, locais de entrega e demais aspectos jurídicos. O grupo das **informações técnicas** é o conteúdo relacionado diretamente com os aspectos técnicos dos objetos das licitações, ou seja, com suas especificações técnicas (características que descrevem o produto a ser adquirido ou o serviço a ser contratado através processo licitatório). Essa classificação possibilitou a criação de um filtro que permitiu a retirada da maioria das informações técnicas dos textos extraídos, tendo em vista que elas não possuem as entidades de interesse para este trabalho. Dessa forma, a anotação nas seções técnicas não foi realizada para manter os *corpora* os mais coesos possíveis. A escolha de um algoritmo de classificação e o processo de construção do modelo usado para realizar a categorização das informações são descritos na Seção 4.2. Também foi feita a divisão dos *corpora* por estado, a saber: Espírito Santo (ES), Minas Gerais (MG), Rio de Janeiro (RJ) e São Paulo (SP).

Na Figura 7 é apresentado um recorte da parte contendo informações jurídicas do edital de pregão eletrônico nº 0002/2019 da Fundação de Amparo à Pesquisa e Inovação do Espírito Santo (FAPES), obtido no Portal de Compras do Estado do Espírito Santo, e na Figura 8 é ilustrado um recorte da parte com informações técnicas deste mesmo edital.

Figura 7 – Exemplo de parte do conteúdo jurídico de um edital.

ESTADO DO ESPÍRITO SANTO
FUNDAÇÃO DE AMPARO À PESQUISA E INOVAÇÃO DO ESPÍRITO SANTO – FAPES

Av. Fernando Ferrari, 1080 – 7º andar – Ed. América Centro Empresarial, Bairro Mata da Praia, CEP.: 29.066-380 – Vitória – ES – email: cpp@fapes.es.gov.br – Tel (27) – 3636-1879/1882

EDITAL DE PREGÃO ELETRÔNICO Nº 0002/2019

A **Fundação de Amparo à Pesquisa e Inovação do Espírito Santo**, doravante denominado **FAPES** realizará licitação, na modalidade "Pregão Eletrônico", sob o critério "menor preço por lote", por meio do site www.compras.es.gov.br, para **AQUISIÇÃO DE SERVIDORES DE ALTO DESEMPENHO**, conforme Processo nº 82520887, devidamente aprovado pela autoridade competente. O Pregão será realizado por Pregoeiro e Equipe de Apoio, designados pela Instrução de Serviço FAPES nº 006/2019, publicada em 08/01/2019, nos termos da Lei 10.520/2002, e subsidiariamente da Lei 8.666/1993, e do Decreto estadual 2.458-R/2010, bem como pelas demais normas pertinentes e condições estabelecidas no presente Edital.

1 - DISPOSIÇÕES PRELIMINARES

1.1 - A sessão pública de processamento do Pregão Eletrônico será realizada por meio da internet, mediante condições de segurança - criptografia e autenticação - em todas as suas fases.

1.2 - Os trabalhos serão conduzidos pelo Pregoeiro designado, por inserção e monitoramento de dados inseridos no aplicativo "Sistema Integrado de Gestão Administrativa – SIGA", no endereço www.compras.es.gov.br, conforme indicado abaixo:

INÍCIO DO ACOLHIMENTO DAS PROPOSTAS: ÀS 08:00 horas do dia 03/07/2019.

Fonte: Espírito Santo (Estado) (2022)

Figura 8 – Exemplo de parte do conteúdo técnico de um edital.

ESTADO DO ESPÍRITO SANTO
FUNDAÇÃO DE AMPARO À PESQUISA E INOVAÇÃO DO ESPÍRITO SANTO – FAPES

3. ESPECIFICAÇÃO DETALHADA DO OBJETO

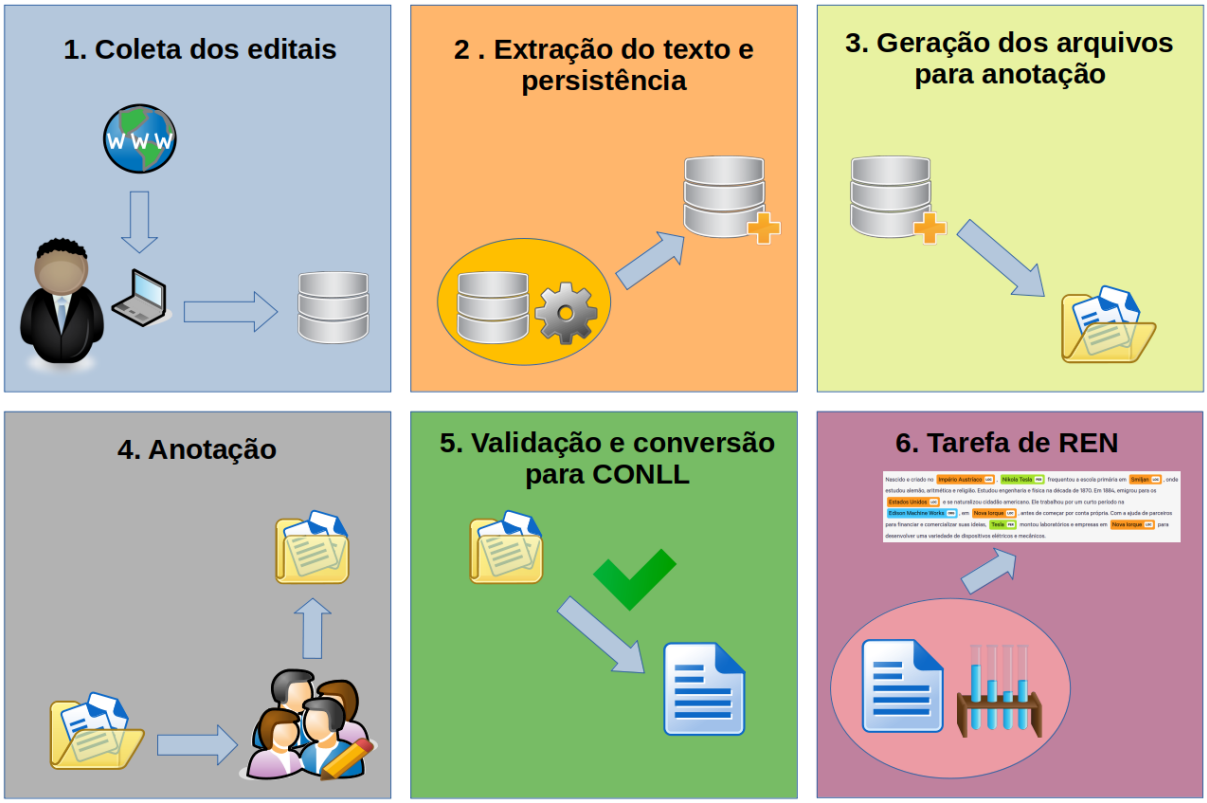
PARTE	DESCRIÇÃO DETALHADA
1	***1.* Processador** **1.1.** Deverá possuir **02 (dois) processadores** com no mínimo **08 (oito) núcleos** de processamento cada um, totalizando, no mínimo **16 (dezesseis) núcleos**; **1.2.** Clock base deverá ser de mínimo de **2.0 GHz** ou Hyper Transport de 4000MHz; **1.3.** Cache L3 de mínimo 11 MB, por processador; **1.4.** Fabricado com tecnologia de produção de no máximo 32 nm; **1.5.** Deverá possuir bus de sistema (QPI/DMI/UMI) de no mínimo **7 GT/s** ou equivalente; **1.6.** Deverá ser projetado para efetuar computação simultânea de **32 bits** e **64 bits**;

Fonte: Espírito Santo (Estado) (2022)

As principais atividades executadas para a criação dos *corpora* e a realização da tarefa

de REN estão ilustradas na Figura 9. Cada uma das atividades foi realizada como segue: **1) Coleta dos editais**. Esta atividade compreendeu a obtenção dos documentos dos editais utilizados para extração dos textos. **2) Extração do texto e persistência**. Nesta atividade, os textos dos editais foram extraídos e persistidos em um banco de dados para serem processados posteriormente. Os documentos estavam no formato *Portable Document Format* (PDF) e após o processo de extração foram armazenados como texto sem formatação. **3) Geração dos arquivos para anotação**. Os textos dos editais armazenados no banco de dados foram recuperados e processados para geração dos arquivos utilizados para os anotadores realizarem as anotações das entidades. **4) Anotação**. Os arquivos gerados na Atividade 3 foram enviados para os anotadores anotarem as entidades (caso existissem) em cada sentença. **5) Validação e conversão para o formato CONLL**. Após a anotação, os arquivos anotados foram devolvidos, validados e convertidos para o formato CONLL. **6) Tarefa de REN**. Os arquivos no formato CONLL gerados na atividade anterior foram utilizados para realização dos experimentos relacionados às tarefas de REN necessárias para a elaboração deste trabalho.

Figura 9 – Principais atividades executadas durante o processo de criação dos *corpora* de editais.



Fonte: Elaborado pelo autor (2022).

Nas seções a seguir são detalhados os processos de coleta (Atividade 1), pré-processamento dos editais (Atividades 2 e 3), anotação (Atividade 4) e, por fim, são apresentadas algumas estatísticas descritivas dos *corpora* criados (Atividade 5) e experimentos realizados para

avaliar o desempenho das melhores configurações dos modelos de REN avaliados no Capítulo 3 nos corpora de editais criados.

4.1 COLETA

Neste trabalho foram considerados somente editais de compras na modalidade pregão eletrônico. No total, 67 editais foram coletados manualmente utilizando os portais dos Governos dos estados do Espírito Santo (ES)[15], Minas Gerais (MG)[16], Rio de Janeiro (RJ)[17] e São Paulo (SP)[18]. Somente editais dos anos de 2020, 2021 e 2022 com a situação de licitação concluída foram recuperados. Na Tabela 6 são detalhados os quantitativos e totais de editais obtidos por estado e por ano de 2020 a 2022.

Tabela 6 – Quantitativo de editais coletados por estado e por ano.

Estado	**2020**	**2021**	**2022**	**Total por estado**
ES	4	4	2	10
MG	2	7	11	20
RJ	6	10	3	19
SP	4	8	6	18
Total geral	16	29	22	67

Fonte: Elaborado pelo autor (2022).

Todos os 67 editais coletados foram processados e anotados de acordo com os procedimentos detalhados na Seção 4.2 e na Seção 4.3.

4.2 PRÉ-PROCESSAMENTO

Os 67 editais recuperados na etapa de Coleta precisaram ser pré-processados para a criação de uma representação estruturada que facilitasse a identificação das informações de interesse para serem enviadas para a etapa de anotação. Inicialmente, foram extraídos os textos brutos dos arquivos PDFs dos editais utilizando as bibliotecas PyPDF2[19] e PDFminer[20]. Essas duas bibliotecas foram usadas de forma complementar para aumentar a cobertura dos conteúdos extraídos dos PDFs, quando uma das bibliotecas apresentava problemas para extrair o conteúdo de um documento, a outra conseguia fazer a extração.

[15] <https://portalsiga.es.gov.br>
[16] <https://www1.compras.mg.gov.br>
[17] <https://www.compras.rj.gov.br>
[18] <https://www.bec.sp.gov.br>
[19] <https://pypi.org/project/PyPDF2>
[20] <https://pypi.org/project/pdfminer>

O conteúdo extraído de cada edital foi dividido em seções e subseções utilizando expressões regulares. Essa divisão foi importante para facilitar a filtragem das informações contidas nas seções técnicas que não são relevantes para o escopo deste trabalho. Para realizar essa filtragem foi desenvolvido um algoritmo de classificação de texto. A classificação das informações dos editais foi feita no momento em que os arquivos para anotação foram gerados, sendo que cada subseção foi classificada em Jurídica (**JUR**) ou Técnica (**TEC**). Somente as subseções classificadas como **JUR** foram utilizadas na geração dos arquivos de anotação. O algoritmo de Florestas Aleatórias, do inglês *Random Forest*, foi utilizado para a classificação das subseções. Esse algoritmo foi escolhido por apresentar os melhores resultados nos experimentos realizados e apresentados no Apêndice A. Posteriormente, somente as subseções classificadas como pertencentes a classe *Jurídica* foram mantidas.

Por fim, as subseções classificadas como jurídicas foram pré-processadas utilizando a ferramenta spaCy[21] para divisão do texto em frases e cada frase fragmentada em palavras. Frases repetidas, ou seja, com o conteúdo exatamente igual, foram descartadas, permanecendo somente com frases distintas.

4.3 PROCESSO DE ANOTAÇÃO

4.3.1 Definição do Formato e Esquema de Anotação

Nesta etapa foi realizado o processo de anotação das entidades nomeadas de interesse deste trabalho. Inicialmente, foi necessário definir o formato dos arquivos utilizados durante a tarefa de anotação e também dos *corpora* produzidos. Para isso, foram escolhidos os formatos de arquivos e esquema de anotação baseados na *Conference on Computational Natural Language Learning* (CoNLL) realizada em 2002 (SANG; MEULDER, 2003). Esse formato foi escolhido por ainda ser muito utilizado em outros *corpora* identificados na literatura (SANTOS et al., 2006; ARAUJO et al., 2018; CASTRO, 2019).

O esquema de anotação das palavras, ou seja, o padrão de rótulos das anotações, usado na CoNLL-2002 e também neste trabalho, é chamando de *Inside–Outside–Beginning* (IOB), onde os prefixos presentes nas etiquetas indicam:

- **B-xxx**. Se presente, indica um início de uma entidade. Por exemplo, se uma palavra pertencer a uma entidade hipotética **Pessoa** e for o início dessa entidade, deverá ser marcado com a etiqueta **B-Pessoa**;

- **I-xxx**. Indica que uma palavra está contida em uma entidade iniciada anteriormente. Ainda com base no exemplo anterior, se a entidade **Pessoa** for composta por mais de

[21] <https://spacy.io>

uma palavra, a primeira palavra deve receber a etiqueta **B-Pessoa** e as demais palavras que compõem a entidade devem ser rotuladas com **I-Pessoa**;

- **O**. Essa etiqueta marca que uma palavra não pertence a nenhuma entidade de interesse.

As frases obtidas dos textos extraídos dos editais, com suas respectivas palavras anotadas, são armazenadas em um arquivo texto. Esse arquivo foi gerado seguindo o formato da CONLL que possui as seguintes características: **(i)** uma palavra por linha; **(ii)** ao lado de cada palavra, separados por um espaço em branco, é colocada a sua etiqueta, seguindo o padrão IOB, indicando à qual entidade ele pertence, ou **O** para indicar que ela não pertence a nenhuma entidade; **(iii)** as frases são separadas por uma linha em branco; e por último, **(iv)** os arquivos gerados nesse formato foram salvos com a extensão **".conll"** para seguir o padrão usado na literatura.

Um fragmento do *corpus* que exemplifica o esquema de anotação e o formato do arquivo detalhado anteriormente é apresentado na Figura 10. Observa-se na figura, que o fragmento possui uma entidade etiquetada como **LOCAL**, composta pelas palavras localizadas nas linhas 1 e 2 do arquivo. Nas linhas 4 e 5, estão as palavras que receberam a etiqueta de **ORGANIZAÇÃO**. Percebe-se também na figura, que as linhas 3 e 6 estão em branco e separam duas sentenças diferentes; e que nas linhas 7, 8, 9 e 12 existem palavras que não foram consideradas partes de nenhuma entidade.

Figura 10 – Exemplo de entidades anotadas nos *corpora* utilizando o esquema IOB e o formato CONLL.

```
 1 ESPÍRITO B-LOCAL
 2 SANTO I-LOCAL
 3
 4 Junta B-ORGANIZACAO
 5 Comercial I-ORGANIZACAO
 6
 7 sanções O
 8 previstas O
 9 no O
10 art. B-ARTIGO
11 81 I-ARTIGO
12 da O
13 Lei B-LEI
14 8.666/1993 I-LEI
15
```

Fonte: Elaborado pelo autor (2022).

4.3.2 Definição das Entidades

Antes de iniciar o processo de anotação das entidades nos documentos editais, foi necessário definir as classes de entidades de interesse que deveriam ser anotadas. A seguir, são apresentadas as entidades consideradas neste trabalho.

- **ORGANIZAÇÃO** – Representa órgãos, empresas, fundações, escolas, dentre outros. Por exemplo, "Agência Estadual de Recursos Hídricos (AGERH)", "ANPD (Autoridade Nacional de Proteção de Dados)", "Banco do Brasil S/A", "Prefeitura Municipal de Amparo/SP".

- **LOCAL** – Representa endereços, cidades e estados. Por exemplo, "Estado do Rio de Janeiro", "Av. Fernando Ferrari, 1080 Goiabeiras, Vitória/ES", "Belo Horizonte/MG".

- **TEMPO** – Menções de expressões de Data e hora em diversos formatos. Por exemplo, "01 de agosto de 2017", "01/02/2021", "15:00h", "07:00h às 22:00h", "08:00 horas do dia 03/12/2012".

- **DECRETO** – Menções de Decretos federais, estaduais e municipais. Por exemplo, "Decreto Estadual nº. 45.902", "Decreto Municipal 6.658/2021", "Decreto federal nº 4.264".

- **ARTIGO** – Menções de Artigos que constam em leis, decretos e portarias. Por exemplo, "ARTIGO 109", "Art. 10", "Art. 25-A", "Art. 28º", "art. 21 e 22".

- **INCISO** – Menções de Incisos que constam em leis, decretos e portarias. Por exemplo, 'INCISO V", "Inc. I", "inciso III e IV", "incisos I a XII".

- **LEI** – Menções de Leis federais, estaduais e municipais. Por exemplo, "Lei Complementar Federal nº 123", "LEI N 11.488", "Lei 23.751/2020", "Lei Complementar estadual 618/2012", "Lei Municipal nº 12.392/05", "Lei federal 8666/1993".

- **PORTARIA** – Menções de Portarias. Por exemplo, "Portaria / SVS 344/98", "Portaria GR nº 248/98", "Portaria SEGER / PGE / SECONT N 049-R/2010".

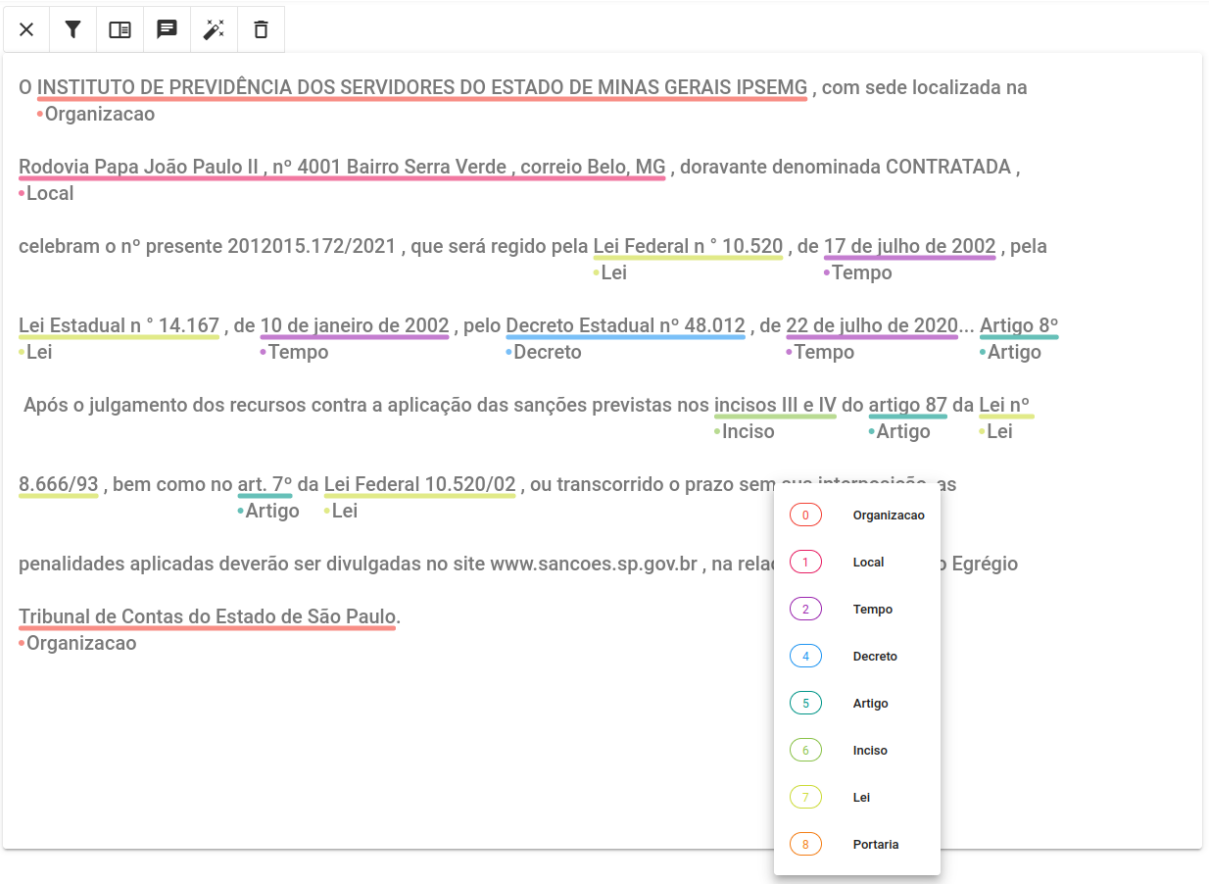
4.3.3 Anotação

Para realizar as anotações nos textos extraídos dos editais, foi formado um grupo com cinco anotadores. Os anotadores são falantes nativos da língua portuguesa, quatro deles com curso superior e um estudante de graduação. Inicialmente, realizou-se uma reunião para explicar o processo de anotação em que foi disponibilizado um documento com o procedimento de instalação, configuração e utilização da ferramenta Doccano[22], que é uma aplicação de código aberto para anotações de textos que auxilia em diversas tarefas de PLN.

[22] <https://doccano.herokuapp.com>

Durante 27 semanas, foi enviado um arquivo por semana para cada anotador. Esse arquivo era enviado por e-mail e possuía no máximo 100 sentenças extraídas dos editais. Cada integrante importava o arquivo usando a ferramenta Doccano, realizava o processo de anotações e, ao final, exportava o arquivo anotado e devolvia o mesmo. Na Figura 11 é apresentado um exemplo de anotação utilizando o Doccano, onde existe uma sentença com várias palavras e algumas delas estão anotadas. As palavras anotadas são marcados pela ferramenta com um grifo – da mesma cor atribuída à etiqueta no momento do cadastro dela na ferramenta – além de informar abaixo da entidade anotada qual foi a etiqueta atribuída. É possível verificar que as palavras *"INSTITUTO", "DE", "PREVIDÊNCIA", "DOS", "SERVIDORES", "DO", "ESTADO", "DE", "MINAS", "GERAIS"* e *"IPSEMG"* foram anotadas juntas – pois formam uma entidade composta por mais de uma palavra – e receberam a etiqueta **ORGANIZAÇÃO**.

Figura 11 – Exemplo da tela de anotação das entidades usando o Doccano.



Fonte: Elaborado pelo autor (2022).

O formato do arquivo exportado, que no Doccano é chamado de JSONL, define o conteúdo do arquivo como uma lista do tipo *JavaScript Object Notation* (JSON). Cada linha do arquivo possui um JSON com dados da sentença anotada e uma lista contendo as posições das palavras e as categorias das entidades atribuídas a essas palavras. Caso a sentença não possua nenhuma palavra que represente uma entidade, a lista de posições estará vazia.

Na Figura 12 é apresentado um exemplo de um arquivo JSONL que contém dois JSONs. O campo *"id"* armazena um código de identificação da sentença exportada; a sentença anotada é armazenada no campo *"data"*; o campo *"label"* possui uma lista de listas que contém as marcações dos limites (posição do início e fim) de cada entidade na sentença e a etiqueta atribuída à entidade.

Figura 12 – Estrutura de um arquivo JSONL gerado usando a ferramenta Doccano.


```
{
"id": 3785,

"data": "43.124 de 29 de setembro de 2011 , Lei nº 6.304 de 28 de agosto de 2012 e
Decreto nº 41.135/2008 , inscrita no CNPJ sob o nº 10.834.118/ 0001-79 , com sede
situada na Avenida Padre Leonel Franca , nº 248 Gávea Rio de Janeiro / RJ.

"label": [[10, 32, "Tempo"], [35, 71, "Lei"], [74, 96, "Decreto"], [167, 229, "Local"]]
}

{
"id": 3840,

"data": "Documento assinado eletronicamente por Dilermando Ribeiro Lima , Diretor
Administravo Financeiro , em 07/01/2021 , às 14:55 , conforme horário ocial de
Brasília , com fundamento nos art. 21º e 22º do Decreto nº 46.730 , de 9 de agosto de
2019 .",

"label": [[102, 112, "Tempo"], [118, 123, "Tempo"], [152, 160, "Local"], [182, 196,
"Artigo"], [200, 242, "Decreto"]]
}
```


Fonte: Elaborado pelo autor (2022).

Após a conclusão das 27 semanas de anotação, todos os arquivos anotados e recebidos da primeira até a última semana foram convertidos para o formato CONLL. Os arquivos foram divididos por estado, e cada *corpus* recebeu como nome a sigla do estado correspondente, sendo que a partir de então, para simplificar, serão referenciados pelas siglas: **ES**, **MG**, **RJ** e **SP** para os *corpora* dos estados do Espírito Santo, Minas Gerais, Rio de Janeiro e São Paulo, respectivamente.

Por fim, foi realizada uma verificação final para revisar as anotações realizadas e corrigir erros, caso fosse preciso. Após as correções e ajustes necessários, foi calculado, com base nos *corpora* originais e os corrigidos, o coeficiente Cohen's kappa (COHEN, 1960) que é uma medida estatística utilizada para calcular a concordância entre os anotadores e um revisor, ou seja, verificar se anotador e revisor atribuíram, ou não, as mesmas etiquetas para os *tokens* do *corpus* analisado. Para efetuar os cálculos dos coeficientes foi utilizada a função *cohen_kappa_score* disponível em[23]. Essa função retorna um valor entre -1 e 1 onde valores próximos de 1 representam um alto grau de concordância e valores menores do que 0 significa falta de concordância. Os valores dos coeficientes gerados foram: ES (0,9620); MG (0,9609); RJ (0,9611) e SP (0,9573). Assim, verifica-se que os coeficientes encontrados denotam uma alta concordância entre os anotadores e o revisor.

[23] <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.cohen_kappa_score>

4.4 ESTATÍSTICAS DOS *CORPORA*

Algumas estatísticas descritivas foram geradas no processo de conversão dos arquivos anotados para o formato CONLL e confecção dos *corpora*. Na Tabela 7 são apresentadas as estatísticas geradas, sendo elas: **(i)** Quantidade de editais processados; **(ii)** Quantidade de sentenças; **(iii)** Média de sentenças por edital; **(iv)** Quantidade de sentenças com entidade; **(v)** Média de sentenças com entidade por edital; **(vi)** Total de palavras; **(vii)** Média de palavras por sentença; **(viii)** Quantidade de entidades anotadas; **(ix)** Média de entidades por sentença com pelo menos uma entidade. Essas estatísticas foram consideradas para cada um dos quatro *corpora* (ES, MG, RJ e SP) e na coluna "Geral" é apresentado o total geral de cada uma das estatísticas. É possível notar que: a diferença na quantidade de editais coletados para cada estado impactou diretamente no total de sentenças, *tokens* e entidades identificados em cada *corpus*.

Tabela 7 – Estatísticas descritivas dos *corpora* por estado.

Medida	**Estados**				
	ES	**MG**	**RJ**	**SP**	**Geral**
Quantidade de editais processados	10	20	19	18	67
Quantidade de sentenças	1.980	7.398	4.403	5.850	19.631
Média de sentenças por edital	198	369,90	231,74	325	293
Quantidade de sentenças com entidade	506	1.326	879	1.043	3.754
Média de sentenças com entidade por edital	50,60	66,30	46,26	57,94	56,03
Total de *tokens*	53.967	187.699	125.630	154.513	521.809
Média de *tokens* por sentença	27,26	25,37	28,53	26,41	26,58
Quantidade de entidades	917	3.054	2.129	2.238	8.338
Média de entidades por sentença com entidade	1,81	2,30	2,42	2,14	2,22

Fonte: Elaborado pelo autor (2022).

A distribuição das entidades anotadas em cada etiqueta é apresentada na Tabela 8. Observa-se que, dentre as 8.338 entidades anotadas, a etiqueta *ORGANIZAÇÃO* possui, de forma geral, a maior quantidade de entidades, seguida pelas etiquetas *ARTIGO*, *LEI* e *TEMPO*, cada uma das três últimas tem mais de 1.300 entidades reconhecidas. Essas quatro etiquetas juntas perfazem um total de 6.632 ($\approx$ 79,54% do total de entidades). Todas as outras etiquetas tiveram um número razoável de entidades reconhecidas, exceto a etiqueta *PORTARIA* que teve uma quantidade pequena, somente com 59 entidades. As linhas da Tabela 8 foram ordenadas considerando a coluna "Geral" (sub-total de entidades reconhecidas), mas é possível notar que: se a ordenação fosse considerada por estado, as posições das etiquetas seriam trocadas, por exemplo, a etiqueta *ARTIGO*, de forma geral, possui mais entidades reconhecidas do que a etiqueta *LEI*, mas no caso específico

do estado ES, a ordem dessas duas etiquetas seria invertida, pois a etiqueta *LEI* possui mais entidades reconhecidas do que a etiqueta *ARTIGO*.

Tabela 8 – Tipos de etiquetas e total de entidades anotadas por estado.

Etiqueta	**Entidades anotadas**				
	ES	**MG**	**RJ**	**SP**	**Geral**
ORGANIZAÇÃO	375	863	702	592	2532
ARTIGO	127	495	304	486	1412
LEI	183	466	278	452	1379
TEMPO	77	640	384	208	1309
LOCAL	72	246	254	241	813
DECRETO	44	255	123	106	528
INCISO	36	77	56	137	306
PORTARIA	3	12	28	16	59

Fonte: Elaborado pelo autor (2022).

4.5 EXPERIMENTOS

Nesta seção são apresentados: **(i)** o ambiente experimental utilizado nos experimentos, incluindo as medidas de avaliação adotadas, o modelo de REN utilizado, a metodologia de avaliação adotada e a configuração do computador utilizado para execução dos experimentos; e **(ii)** os resultados dos experimentos realizados nos *corpora* de editais criados divididos por estado.

4.5.1 Ambiente Experimental

Medidas de avaliação. Assim como no Capítulo 3, todas as medidas de avaliação utilizadas nos experimentos foram calculadas usando o *script* de avaliação disponibilizado em[24]. Também foram adotadas as medidas de avaliação micro (Precisão, Cobertura, e Medida-F) em nível de entidade, considerando somente correspondências exatas.

Modelo de Classificação de EN. Neste experimento foram considerados os *embeddings* tradicionais do Word2vec Skip-gram e Glove, em conjunto com os *embeddings* contextuais extraídos do modelo BERT nas arquiteturas *base* e *large*. Essas representações foram utilizadas por apresentarem os melhores resultados nos experimentos realizados e descritos no Capítulo 3. Além disso, o modelo BiLSTM-CRF implementado usando o *framework* FLAIR também foi utilizado, seguindo as mesmas configurações apresentadas na Subseção 3.4.1.

[24] <https://www.clips.uantwerpen.be/conll2002/ner/>

Metodologia de Avaliação. Para avaliar o desempenho dos modelos nos corpora de editais, foi utilizada a metodologia de validação cruzada em nível de corpus com 4 subconjuntos, no qual: **(i)** os *corpora* foram disponibilizados usando uma lista e organizados em ordem alfabética: ES, MG, RJ e SP. **(ii)** para compor uma rodada de experimentos, cada um dos *corpora* recebia uma função de teste, validação e treinamento. Por exemplo: Iniciando a rodada com o *corpus* ES, o mesmo recebia a função de teste; na sequência, o de MG era utilizado como validação e os dois restantes (RJ e SP) usados como treinamento. Na segunda rodada, o *corpus* MG recebia a função de teste; RJ, validação e os dois restantes (SP e ES) recebiam a função de treinamento. E assim sucessivamente até que todos os *corpora* fossem usados uma vez para teste e validação, e três vezes para treinamento. Rotacionando os *corpora* dessa maneira, foi possível obter quatro conjuntos diferentes para utilização nos experimentos.

Configuração do Equipamento. Todos os experimentos foram executados em um computador com processador Intel Core i5, com 32 Gigabytes (GB) de memória RAM, GPU Nvidia GeForce RTX 3080 com 12 GB e com Sistema Operacional Ubuntu 20.04.

4.5.2 Resultados dos Experimentos

Nesta subseção são apresentados os resultados dos experimentos realizados com base nos modelos e *corpora* de editais criados. Na Tabela 9 são mostrados os resultados com base na medida-F dos 32 experimentos realizados, considerando a combinação dos *word embeddings* tradicionais e contextualizados. Foram destacados em negrito os melhores valores e, para auxiliar no destaque do modelo que obteve o melhor desempenho, os segundos melhores valores foram marcados em negrito e adicionados de um asterisco. As colunas Média e Desvio Padrão (DP) também foram adicionadas para auxiliar na escolha do melhor modelo, tendo em vista que os melhores valores ficaram dispersos entre alguns modelos e *corpus*. Todos os modelos foram baseados na arquitetura BiLSTM-CRF, tendo como primeira camada uma representação multidimensional baseada na utilização de *word embeddings*: tradicionais, contextualizados ou na combinação dos dois.

Observa-se na Tabela 9 que os resultados obtidos com o uso dos *embeddings* contextualizados isoladamente foram, em geral, melhores que os resultados dos *embeddings* tradicionais utilizados sem combinação. Esse comportamento é similar aos resultados obtidos no Capítulo 3 utilizando as bases de dados da literatura. Analisando os cenários onde os *embeddings* extraídos do $BERT_{Base}$ e $BERT_{Large}$ foram utilizados sozinhos e os cenários onde eles foram combinados com o *embedding* do Glove, verifica-se que houve um empate, sendo que dos 8 cenários (4 *corpora* $\times$ 2 modelos de *embeddings*) analisados, os *embeddings* $BERT_{Base}$ e $BERT_{Large}$ usados sozinhos foram melhores em 4 cenários (50%) e a combinação com o Glove foi melhor nos outros 4 cenários.

Tabela 9 – Resultados (%) dos experimentos das combinações dos *embeddings* avaliados com base na Medida-F. Os dois melhores resultados em cada coluna estão destacados em negrito. O segundo melhor resultado também está marcado com um asterisco.

Tradicional	Contextualizado	ES	MG	RJ	SP	Média	DP
-	$BERT_{Base}$	**71,87***	78,63	**75,82**	76,07	75,60	**2,42**
	$BERT_{Large}$	71,58	81,00	73,28	77,68	75,88	3,70
Glove	-	58,54	74,22	70,62	73,27	69,16	6,27
	$BERT_{Base}$	68,99	81,17	74,54	**79,16**	75.97	4,69
	$BERT_{Large}$	70,18	**83,00**	75,28	76,22	**76,17***	4,56
Word2vec Skip-gram	-	57,20	79,32	70,61	74,71	70,46	8,25
	$BERT_{Base}$	**72,50**	**81,61***	**75,64***	**78,62***	**77,09**	**3,39***
	$BERT_{Large}$	70,63	81,55	74,32	78,14	76,16	4,09

Fonte: Elaborado pelo autor (2022).

Ao considerar os cenários em que os *embeddings* extraídos do $BERT_{Base}$ e $BERT_{Large}$ foram utilizados sozinhos e os cenários onde eles foram combinados com o *embedding Word2vec Skip-gram*, que também perfaz um total de 8 cenários, observa-se que a combinação deles com o *Word2vec Skip-gram* obteve o melhor resultado em 6 dos 8 cenários (75%).

Com base nos resultados experimentais obtidos, conclui-se que: **(i)** a combinação de *embeddings* tradicionais e contextuais levou a melhores resultados na maioria dos cenários avaliados, especialmente usando *Word2vec Skip-gram*; e **(ii)** a arquitetura usando $BERT_{Base}$ com o *Word2vec Skip-gram* (BiLSTM-CRF $BERT_{Base}$-W2V) obteve o melhor desempenho geral considerando todos os *corpora* analisados.

Na Tabela 10 são apresentados os resultados obtidos usando o modelo BiLSTM-CRF $BERT_{Base}$-W2V (melhor modelo) detalhados por entidades, considerando Precisão (P), Cobertura (C) e Medida-F (F1) para cada um dos quatro *corpora*: ES, MG, RJ e SP. Conforme destacado na tabela, a entidade *LEI* foi a que obteve os melhores resultados relacionados com os *corpora* ES e RJ; *DECRETO* foi destaque no *corpus* MG e *INCISO* obteve a maior Cobertura e Medida-F considerando o *corpus* SP.

Para todos os *corpora* analisados, as entidades *ARTIGO* e *LEI* alcançaram resultados superiores a 94% para Precisão e Cobertura; e acima de 95% de Medida-F. A entidade *DECRETO* apresentou desempenho similar aos desempenhos das entidades *ARTIGO* e *LEI*, exceto para o *corpus* ES, o que pode ser explicado pela quantidade inferior de exemplos dessa entidade em relação aos outros *corpora*. A entidade *INCISO*, de maneira geral, também obteve bons resultados para todos os *corpora*.

Tabela 10 – Resultados dos experimentos da arquitetura que apresentou o melhor desempenho (BiLSTM-CRF $BERT_{Base}$-W2V) avaliados com base na Precisão (%), Cobertura (%) e Medida-F (F1) por tipo de entidade nos *corpora* divididos por estado. Os maiores resultados em cada *corpus* estão destacados em negrito.

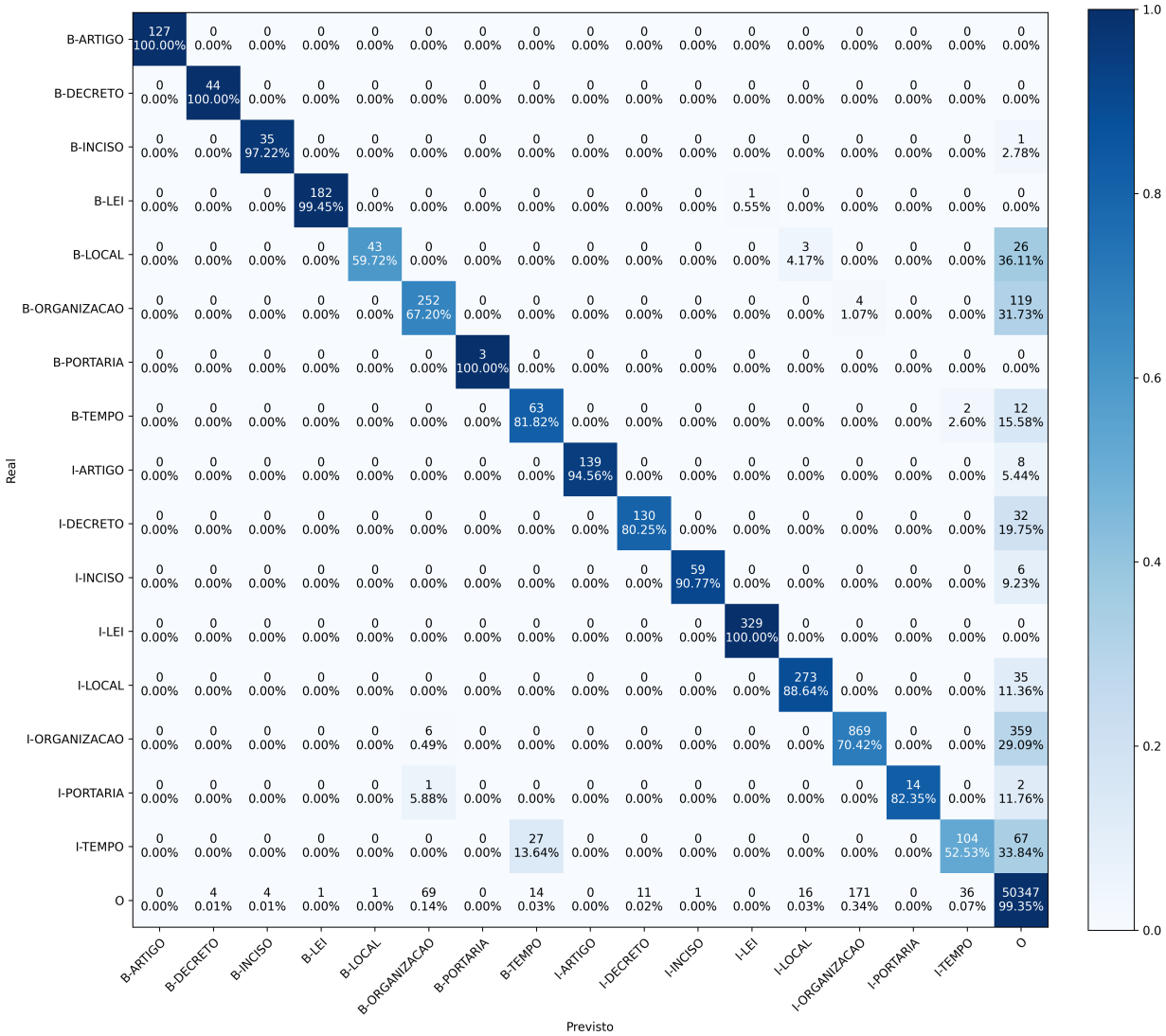
Entidade	**ES**			**MG**			**RJ**			**SP**		
	P	**C**	**F1**	**P**	**C**	**F1**	**P**	**C**	**F1**	**P**	**C**	**F1**
ARTIGO	98,43	98,43	98,43	97,38	97,78	97,58	94,29	97,70	95,96	96,14	97,33	96,73
DECRETO	25,00	27,27	26,09	**100,00**	**100,00**	**100,00**	96,00	97,56	96,77	91,89	96,23	94,01
INCISO	87,18	94,44	90,67	80,65	97,40	88,24	84,38	96,43	90,00	95,74	**98,54**	**97,12**
LEI	**99,45**	**99,45**	**99,45**	95,01	98,07	96,52	**98,57**	**98,92**	**98,74**	**97,32**	96,24	96,77
LOCAL	70,45	43,06	53,45	74,63	40,65	52,63	55,47	53,94	54,69	32,00	29,88	30,90
ORGANIZAÇÃO	67,99	59,47	63,44	66,67	56,78	61,33	55,49	40,31	46,70	58,52	58,61	58,57
PORTARIA	66,67	66,67	66,67	19,64	91,67	32,35	95,24	71,43	81,63	72,22	81,25	76,47
TEMPO	39,42	53,25	45,30	82,09	90,94	86,29	87,83	95,83	91,66	84,55	89,42	86,92

Fonte: Elaborado pelo autor (2022).

Em resumo, as entidades *ARTIGO*, *DECRETO*, *INCISO* e *LEI* foram as que obtiveram os melhores resultados. Podendo-se concluir que o modelo teve mais facilidade em identificar essas entidades em comparação às outras. Por outro lado, as entidades das classes *ORGANIZAÇÃO* e *LOCAL* representam um desafio para o modelo, como pode ser visto pelos resultados obtidos em todos os corpora. O baixo número de exemplos de entidades da classe *PORTARIA* impactaram diretamente na capacidade do modelo em reconhecer entidades dessa categoria.

Na Figura 13, Figura 14, Figura 15 e Figura 16 são apresentadas, respectivamente, as matrizes de confusão baseadas nos *corpora* ES, MG, RJ e SP e no modelo BiLSTM-CRF $BERT_{Base}$-W2V. De acordo com essas matrizes de confusão, percebe-se que o modelo, de forma geral, conseguiu diferenciar bem as etiquetas que formam as entidades. Entretanto, quando houve alguma confusão foi com a etiqueta *O*, que representa *tokens* que não pertencem à nenhuma entidade. Por exemplo, na Figura 13 a etiqueta *B-ORGANIZAÇÃO* foi confundida com a *O* 119 vezes, representando 31,73% de erro. Conforme a matriz de confusão do *corpus* ES, as etiquetas que *B-LOCAL*, *B-ORGANIZAÇÃO*, *I-ORGANIZAÇÃO* e *I-TEMPO* obtiveram uma taxa de erro superior a 29%; e *B-TEMPO* e *I-DECRETO* acima de 15%, indicando certa dificuldade do modelo em identificar corretamente essas etiquetas.

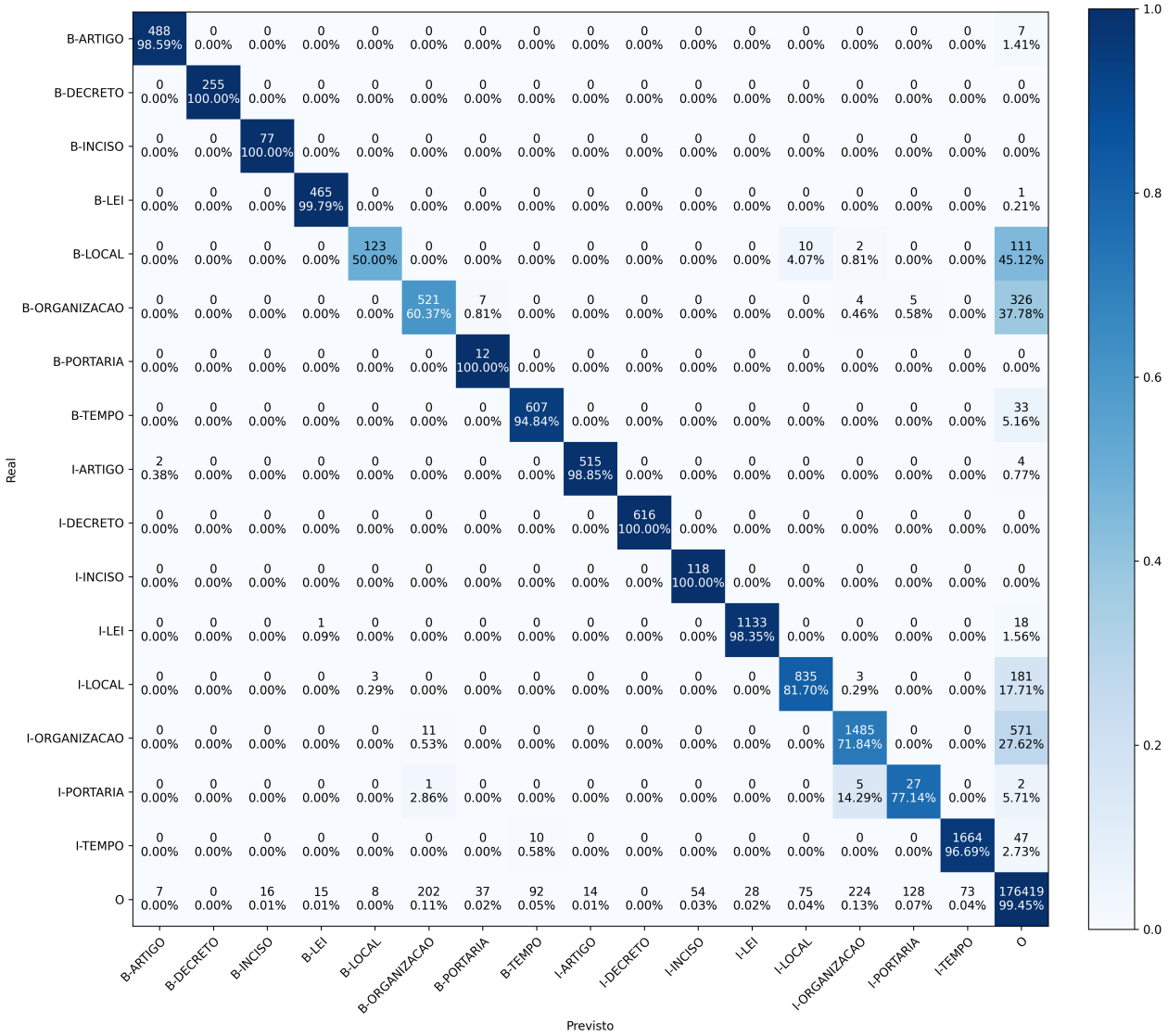
Figura 13 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Base}$-W2V no *corpus* do ES.



Fonte: Elaborado pelo autor (2022).

Considerando a matriz de confusão relacionada ao *corpus* MG, as entidades que obtiveram maior taxa de confusão foram: *B-LOCAL* (45,12%), *B-ORGANIZAÇÃO* (37,78%), *I-LOCAL* (17,71%) e *I-ORGANIZAÇÃO* (27,62%). Observa-se nessa matriz que as maiores taxas de confusão, assim como na matriz de confusão do ES, foram em relação à etiqueta *O*, porém, uma quantidade menor de etiquetas obtiveram taxas de confusão elevadas.

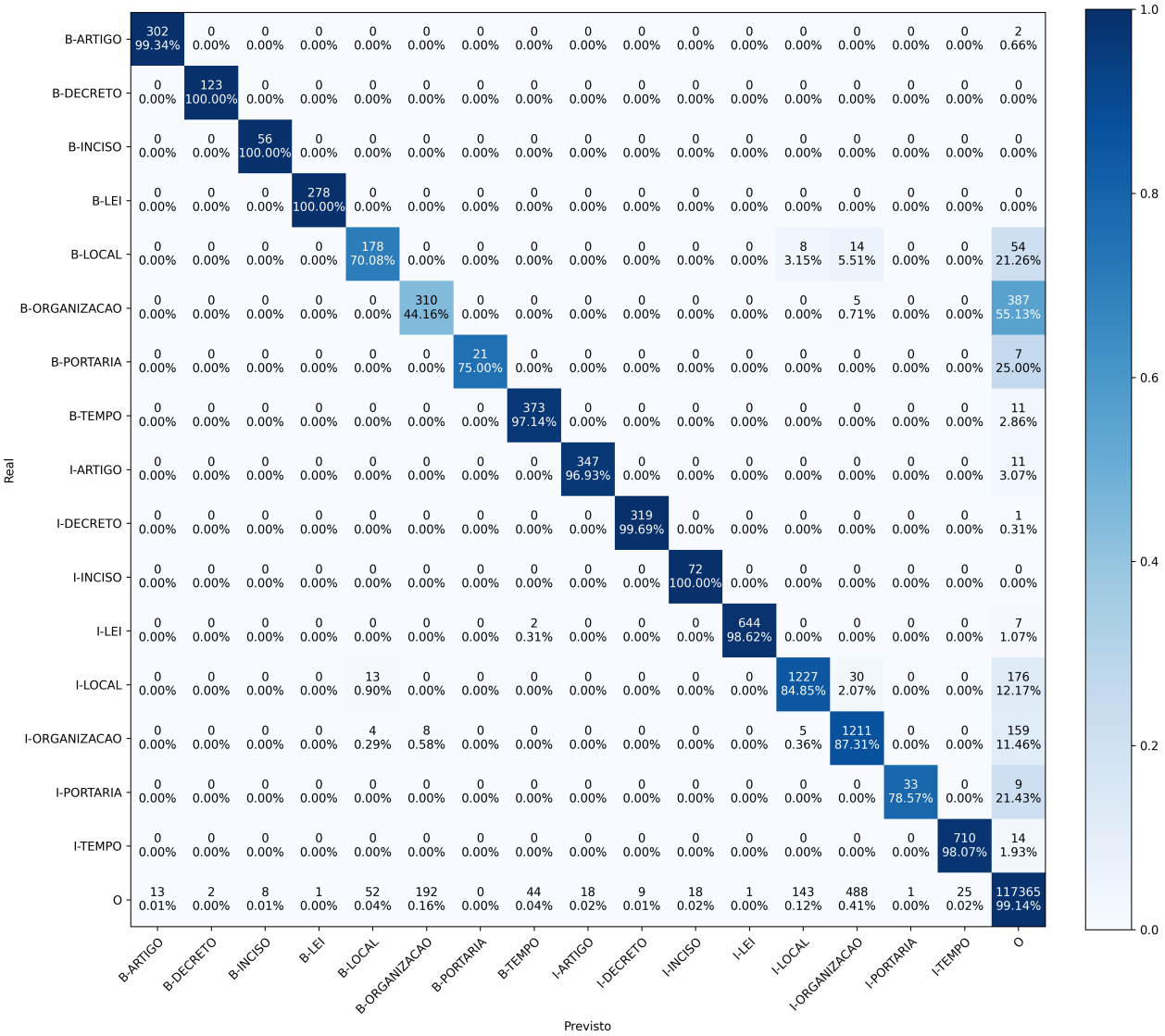
Figura 14 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Base}$-W2V no *corpus* de MG.



Fonte: Elaborado pelo autor (2022).

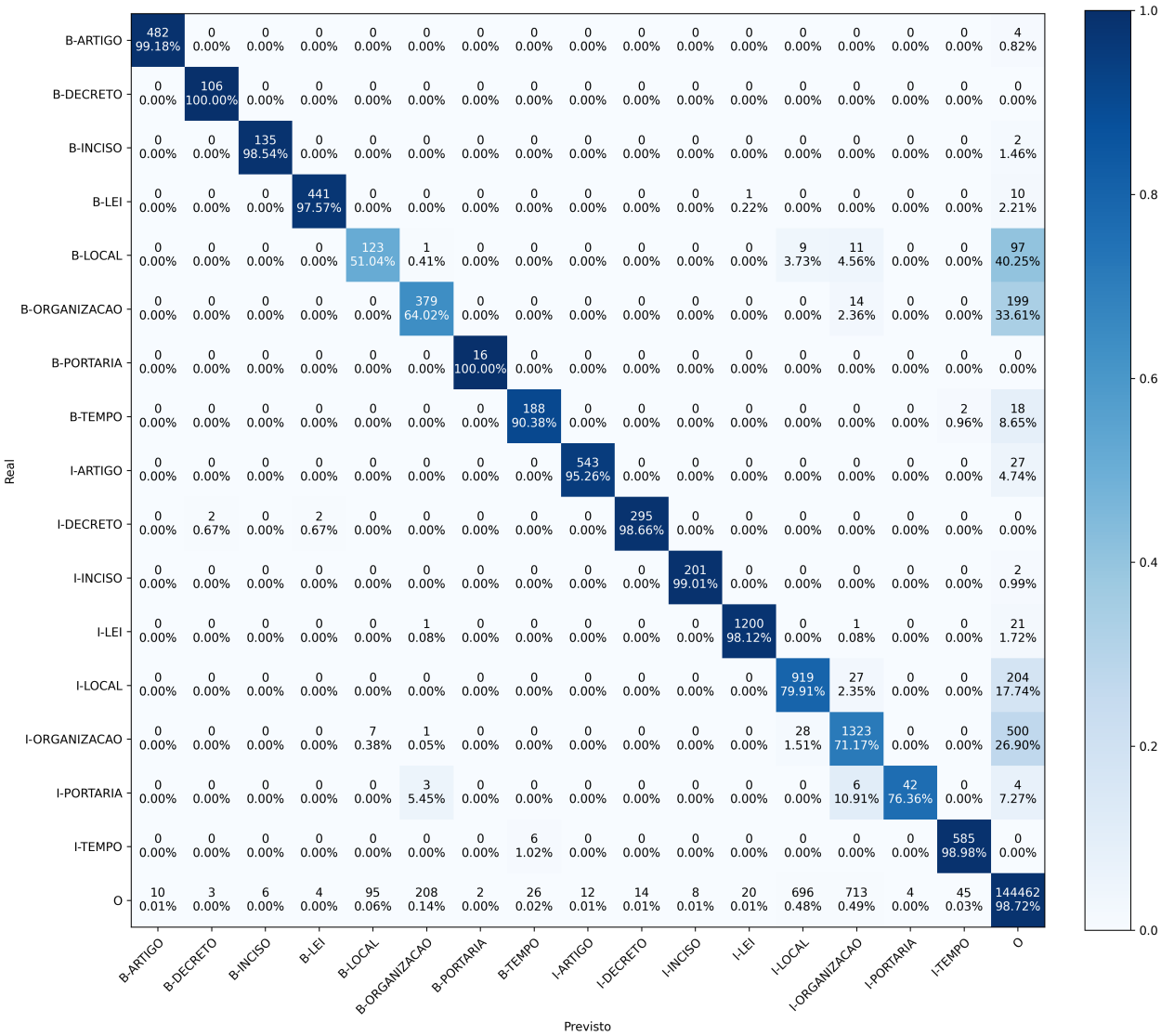
Com relação ao *corpus* do RJ, as etiquetas que apresentaram as maiores taxas de confusão foram *B-LOCAL* e *I-PORTARIA*, com pouco mais de 21% cada; *B-PORTARIA* com 25% e *B-ORGANIZAÇÃO* com 55,13% cuja taxa de confusão foi a maior entre todas as etiquetas dos quatro *corpora*. Na matriz de confusão do *corpus* de SP nota-se que as etiquetas com as maiores taxas de confusão foram: *B-LOCAL* (40,25%), *I-LOCAL* (17,74%), *B-ORGANIZAÇÃO* (33,61%) e *I-ORGANIZAÇÃO* (26,90%).

Figura 15 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Base}$-W2V no *corpus* do RJ.



Fonte: Elaborado pelo autor (2022).

Figura 16 – Matriz de Confusão da arquitetura BiLSTM-CRF $BERT_{Base}$-W2V no *corpus* de SP.



Fonte: Elaborado pelo autor (2022).

Adicionalmente foram calculadas a média e o desvio padrão com base nas taxas de acerto para cada etiqueta em cada um dos quatro *corpora*. As taxas de acerto são encontradas nas diagonais principais das quatro matrizes de confusão (Figuras 13, 14, 15 e 16) e estão destacadas na cor azul com mais ou menos intensidade. Os valores da média e desvio padrão foram, respectivamente, para cada *corpus*: **(i)** ES (86,13% e 15,03%); **(ii)** MG (89,86% e 15,37%); **(iii)** RJ (89,94% e 14,93%); **(iv)** SP (89,23% e 14,65%).

Em resumo, é possível concluir que: **(i)** as etiquetas *B-ORGANIZAÇÃO*, *I-ORGANIZAÇÃO*, *B-LOCAL* e *I-LOCAL* apresentaram altas taxas de confusão em todos os *corpora*; **(ii)** o desvio padrão, considerando a taxa de acerto, foi elevado para todos os *corpora* (entre 14,65% e 15,37%); e **(iii)** todos os *corpora* obtiveram uma boa média em relação às taxas de acerto (superiores a 89%), exceto o *corpus* do ES cuja média foi de 86,13%.

5 CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Neste trabalho foi investigada a aplicação de diferentes arquiteturas baseadas em redes neurais profundas para o reconhecimento de entidades nomeadas em documentos de editais de compras escritos em português do Brasil. Para isso, foi desenvolvido um modelo que utiliza uma arquitetura baseada em uma rede *Bidirectional Long Short-Term Memory* (BiLSTM) com o algoritmo *Conditional Random Field* (CRF), chamada de *BiLSTM-CRF*, usada em conjunto com representações baseadas em *word embeddings* tradicionais (Glove e Word2vec) e contextuais (BERT, Elmo e Flair). Inicialmente, os modelos de *word embeddings* foram avaliados individualmente e de forma combinada utilizando os *corpora* do Harem (Total e Seletivo), Paramopama e LeNER. Todos os resultados foram analisados com base nas medidas de Precisão, Cobertura, Medida-F e também em matrizes de confusão.

Com base nos resultados experimentais obtidos, pode-se concluir que o modelo BiLSTM-CRF $BERT_{Large}$-W2V, construído a partir da combinação dos *embeedings* extraídos do $BERT_{Large}$ e do *Word2Vec*, alcançou o melhor desempenho entre os modelos avaliados. Além disso, esse modelo obteve resultados competitivos com outras abordagens que alcançaram o estado da arte para o idioma português considerando apenas o *corpus* HAREM em ambos os cenários (Total e Seletivo).

Por fim, foram construídos quatro *corpora*, conforme descrito no Capítulo 4, com base nos editais de compras para licitações. Esses *corpora* passaram por um processo de revisão e, em seguida, foram realizados experimentos utilizando modelos baseados em uma BiLSTM-CRF e na combinação dos *word embeddings* tradicionais do Glove e do Word2vec com as representações contextuais extraídas do modelo BERT (nas versões *Base* e *Large*). Com base nos resultados destes experimentos, foi selecionada a arquitetura que combina o $BERT_{Base}$ com o *Word2vec Skip-gram* (BiLSTM-CRF $BERT_{Base}$-W2V) dado que obteve o melhor desempenho geral, considerando todos os *corpora* analisados.

5.1 CONTRIBUIÇÕES

Foi realizado um estudo acerca da viabilidade de combinar *word embeddings* tradicionais e contextuais para reconhecimento de entidades nomeadas em português. Foram avaliados os *word embeddings* tradicionais Glove e Word2vec Skip-gram e os contextuais gerados a partir do BERT (versões *Base* e *Large*), Elmo e Flair. Uma arquitetura baseada em um BiLSTM-CRF foi empregada como modelo de classificação. Experimentos foram realizados considerando o Primeiro Harem e o MiniHarem nos cenários Total e Seletivo, Paramopama combinado com a segunda versão do Harem e LeNER-BR. Os resultados experimentais obtidos demonstraram que a estratégia de combinação de *embeddings* levou a um melhor desempenho na maioria dos cenários avaliados. Em particular, a combinação do Word2vec

Skip-gram com o BERT *Large* apresentou resultados competitivos com sistemas NER do estado da arte. Detalhes sobre o estudo estão dispostos no Capítulo 3 e publicados em um artigo, conforme citado na Seção 5.3.

Outra contribuição foi o aproveitamento do estudo realizado com os *corpora* disponíveis na literatura para direcionar outra investigação sobre a utilização da combinação dos *word embeddings* aplicada à tarefa de REN em documentos de editais de compras dos estados do Espírito Santo (ES), Minas Gerais (MG), Rio de Janeiro (RJ) e São Paulo (SP). Foi necessária a criação de um *corpus*, baseado em editais de compras, para cada um dos estados citados anteriormente. Após a construção dos *corpora*, os mesmos foram submetidos a vários experimentos utilizando os modelos baseados na estratégia de combinação de *embeddings*. A análise dos resultados obtidos levou à conclusão de que essa estratégia também conseguiu bons resultados ao ser aplicada aos documentos de editais de compras dos estados ES, MG, RJ e SP. Os detalhes dessa análise foram apresentados no Capítulo 4. Os *corpora* confeccionados, assim como todos os códigos e artefatos produzidos nesse trabalho, estão disponíveis em[25].

5.2 TRABALHOS FUTUROS

Como trabalhos futuros, pretende-se: **(i)** Expandir os *corpora* e experimentos para editais de compras de outros estados; **(ii)** Investigar novas arquiteturas e configurações para a criação de outros modelos de classificação de entidades nomeadas; e **(iii)** Desenvolver uma ferramenta que aplique o melhor modelo desenvolvido no Capítulo 4 para reconhecer e marcar as entidades em documentos de editais de compras, facilitando a identificação dessas entidades pelos usuários da ferramenta.

5.3 PUBLICAÇÃO DECORRENTE DO TRABALHO

Parte dos resultados obtidos neste trabalho foram publicados no seguinte artigo:

- SILVA, M. G. da; OLIVEIRA, H. T. A. de. Combining word embeddings for portuguese named entity recognition. In: *Computational Processing of the Portuguese Language*. Cham: Springer International Publishing, 2022. v. 13208, p. 198–208. ISBN 978-3-030-98305-5.

[25] <https://github.com/messias077/ren_editais>

REFERÊNCIAS

AKBIK, A. et al. FLAIR: An easy-to-use framework for state-of-the-art NLP. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 54–59.

AKBIK, A.; BLYTHE, D.; VOLLGRAF, R. Contextual string embeddings for sequence labeling. In: *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, 2018. p. 1638–1649.

ALBUQUERQUE, H. O. et al. Ulyssesner-br: A corpus of brazilian legislative documents for named entity recognition. In: SPRINGER. *International Conference on Computational Processing of the Portuguese Language*. [S.l.], 2022. p. 3–14.

ALLES, V. J. *Construção de um corpus para extrair entidades nomeadas do Diário Oficial da União utilizando aprendizado supervisionado*. 2018. 60 p. Dissertação (Mestrado em Engenharia Elétrica) — Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, 2018.

AMARAL, D. O. F. d. *O reconhecimento de entidades nomeadas por meio de conditional random fields para a língua portuguesa*. 2013. 100 p. Dissertação (Mestrado em Ciência da Computação) — Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, 2013.

AMORIM, V. A. J. d. Licitações e contratos administrativos: teoria e jurisprudência - 2a. ed. *Brasília: Senado Federal, Coordenação de Edições Técnicas*, 2018.

ARAUJO, P. H. L. de et al. Lener-br: a dataset for named entity recognition in brazilian legal text. In: SPRINGER. *International Conference on Computational Processing of the Portuguese Language*. [S.l.], 2018. p. 313–323.

ARKHIPOV, M. Y.; BURTSEV, M. S. et al. Application of a hybrid bi-lstm-crf model to the task of russian named entity recognition. In: SPRINGER. *Conference on Artificial Intelligence and Natural Language*. [S.l.], 2017. p. 91–103.

ATHAVALE, V. et al. Towards deep learning in hindi ner: An approach to tackle the labelled data scarcity. *arXiv preprint arXiv:1610.09756*, 2016.

BARALIS, E. et al. Multi-document summarization based on the yago ontology. *Expert Syst. Appl.*, v. 40, n. 17, p. 6976–6984, 2013.

BRANDÃO, F. T. *Eficiência na gestão das compras públicas: um estudo de caso da modalidade pregão eletrônico nos processos licitatórios do Instituto Federal de Educação, Ciência e Tecnologia do Sertão Pernambucano*. 2016. 157 p. Dissertação (Mestrado Profissional em Gestão de Processos Institucionais) — Centro de Ciências Humanas, Letras e Artes, Universidade Federal do Rio Grande do Norte, Natal, 2016.

BRASIL. *Constituição (1988). Constituição da República Federativa do Brasil. Brasília, DF: Senado*. 1988. Disponível em: <https://www2.camara.leg.br/atividade-legislativa/legislacao/constituicao1988>. Acesso em: 28 abr. 2022.

BRASIL. *Lei 8.666, de 21 de Junho de 1993. Regulamenta o art. 37, inciso XXI, da Constituição Federal, institui normas para licitações e contratos da Administração Pública e dá outras providências*. 1993. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/l8666cons.htm>. Acesso em: 23 mai. 2022.

BRASIL. *Lei de Licitações e Contratos Administrativos*. 2021. Disponível em: <http://www.planalto.gov.br/ccivil_03/_Ato2019-2022/2021/Lei/L14133.htm>. Acesso em: 26 mai. 2022.

CARMO, P. et al. Aprendizado de máquina em ações de controle no tribunal de contas do estado do maranhão. *VII Jornada de Informática do Maranhão*, 2018.

CARVALHO, W. S. *Reconhecimento de entidades mencionadas em português utilizando aprendizado de máquina*. 2012. 80 p. Dissertação (Mestrado em Ciência da Computação) — Universidade de São Paulo, São Paulo, 2012.

CASTRO, P. V. Q. d. *Aprendizagem profunda para reconhecimento de entidades nomeadas em domínio jurídico*. 2019. 125 p. Dissertação (Mestrado em Ciência da Computação) — Instituto de Informática, Universidade Federal de Goiás, Goiânia, 2019.

CASTRO, P. V. Q. de; SILVA, N. F. F. da; SOARES, A. da S. Contextual representations and semi-supervised named entity recognition for portuguese language. CEUR-WS.org, v. 2421, p. 411–420, 2019.

CASTRO, P. V. Quinta de; SILVA, N. Félix Felipe da; SOARES, A. da S. Portuguese named entity recognition using lstm-crf. In: VILLAVICENCIO, A. et al. (Ed.). *Computational Processing of the Portuguese Language*. [S.l.]: Springer International Publishing, 2018. p. 83–92. ISBN 978-3-319-99722-3.

COHEN, J. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, Sage Publications Sage CA: Thousand Oaks, CA, v. 20, n. 1, p. 37–46, 1960.

DAS, A.; GARAIN, U. Crf-based named entity recognition@ icon 2013. *arXiv preprint arXiv:1409.8008*, 2014.

DEVLIN, J. et al. 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 4171–4186.

(ESTADO), E. S. *Portal de Compras do Governo do Estado do Espírito Santo*. 2022. Disponível em: <https://compras.es.gov.br/>. Acesso em: 26 mai. 2022.

FERRI, J. *Abordagem modular baseada em dicionário para reconhecimento de entidades nomeadas através de associação aproximada*. 2016. 50 p. Dissertação (Mestrado em Informática) — Universidade Federal do Paraná, Curitiba, 2016.

FONSECA, E. B. et al. Reconhecimento de entidades nomeadas para o português usando o opennlp. *Anais do ENIAC 2015, 2015, Brasil.*, 2015.

GARCIA, G. L.; NETO, J. B. C. Avaliação de superfaturamento em licitações públicas. *Revista Eletrônica eF@tec*, v. 8, n. 1, p. 10–10, 2018.

GERS, F. A.; SCHMIDHUBER, J. Recurrent nets that time and count. In: IEEE. *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*. [S.l.], 2000. v. 3, p. 189–194.

GERS, F. A.; SCHMIDHUBER, J.; CUMMINS, F. Learning to forget: Continual prediction with lstm. *Neural computation*, MIT Press, v. 12, n. 10, p. 2451–2471, 2000.

GRISHMAN, R.; SUNDHEIM, B. Message understanding conference-6: A brief history. In: *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*. USA: Association for Computational Linguistics, 1996. (COLING '96), p. 466–471.

HARTMANN, N. et al. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In: *Symposium in Information and Human Language Technology (STIL)*. [S.l.: s.n.], 2017.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural computation*, MIT Press, v. 9, n. 8, p. 1735–1780, 1997.

HOUDT, G. V.; MOSQUERA, C.; NÁPOLES, G. A review on the long short-term memory model. *Artificial Intelligence Review*, Springer, v. 53, n. 8, p. 5929–5955, 2020.

ISHIKAWA, L.; ALENCAR, A. C. de. Compliance inteligente: o uso da inteligência artificial na integridade das contratações públicas. *Revista de Informação Legislativa*, Senado Federal, v. 57, n. 225, p. 83–98, 2020.

JIANG, J. Information extraction from text. In: _____. *Mining Text Data*. Boston, MA: Springer US, 2012. p. 11–41. ISBN 978-1-4614-3223-4.

JIANG, R.; BANCHS, R. E.; LI, H. Evaluating and combining name entity recognition systems. In: *Proceedings of the Sixth Named Entity Workshop*. [S.l.: s.n.], 2016. p. 21–27.

JIN, Y. et al. Lstm-crf neural network with gated self attention for chinese ner. *IEEE Access*, IEEE, v. 7, p. 136694–136703, 2019.

JULIÃO, D. V. S. *Raciocínio baseado em casos na confecção de termos de referência para contratações públicas*. 2016. 104 p. Dissertação (Mestrado em Ciência da Computação) — Centro de informática, Universidade Federal de Pernambuco, Recife, 2016.

JÚNIOR, C. M. et al. Paramopama: a brazilian-portuguese corpus for named entity recognition. *Encontro Nac. de Int. Artificial e Computacional*, 2015.

KARIYADO, M. Y. *Proposta e aplicação de uma sistemática baseada na gestão do conhecimento para a melhoria do processo de compras públicas*. 2016. 140 p. Dissertação (Mestrado em Gestão de Organizações e Sistemas Públicos.) — Universidade Federal de São Carlos, São Carlos, 2016.

KONKOL, M.; KONOPÍK, M. Crf-based czech named entity recognizer and consolidation of czech ner research. In: SPRINGER. *International conference on text, speech and dialogue*. [S.l.], 2013. p. 153–160.

LAFFERTY, J.; MCCALLUM, A.; PEREIRA, F. C. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In: ACM. *18th International Conference on Machine Learning 2001 (ICML 2001)*. [S.l.], 2001. p. 282–289.

LAMPLE, G. et al. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*, 2016.

LI, J. et al. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, IEEE, 2020.

LI, M. et al. An unsupervised learning approach for ner based on online encyclopedia. In: SPRINGER. *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data*. [S.l.], 2019. p. 329–344.

LIU, X. et al. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, 2015. p. 912–921.

MANNING, C. D.; RAGHAVAN, P.; SCHÜTZE, H. *An Introduction to Information Retrieval*. [S.l.]: Cambridge University Press Cambridge, 2009.

MERVIN, R. An overview of question answering system. *International Journal Of Research In Advance Technology In Engineering (IJRATE)*, v. 1, p. 11–14, 2013.

MIKOLOV, T. et al. Distributed representations of words and phrases and their compositionality. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*. Red Hook, NY, USA: Curran Associates Inc., 2013. (NIPS'13), p. 3111–3119.

NETO, J. F. d. S. *Reconhecimento de entidades nomeadas para o português usando redes neurais*. 2019. 93 p. Dissertação (Mestrado em Ciência da Computação) — Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, 2019.

OLIVEIRA, H. et al. Assessing shallow sentence scoring techniques and combinations for single and multi-document summarization. *Expert Systems with Applications*, v. 65, p. 68–86, 2016. ISSN 0957-4174.

PANCHENDRARAJAN, R.; AMARESAN, A. Bidirectional lstm-crf for named entity recognition. In: *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation*. [S.l.: s.n.], 2018.

PELLUCCI, P. R. S. et al. Utilização de técnicas de aprendizado de máquina no reconhecimento de entidades nomeadas no português. *e-xacta*, v. 4, n. 1, 2011.

PENNINGTON, J.; SOCHER, R.; MANNING, C. GloVe: Global vectors for word representation. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, 2014. p. 1532–1543.

PETERS, M. E. et al. Deep contextualized word representations. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, 2018. p. 2227–2237.

PIROVANI, J.; OLIVEIRA, E. Portuguese named entity recognition using conditional random fields and local grammars. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. [S.l.: s.n.], 2018.

PIROVANI, J. P. C. *CRF+ LG: uma abordagem híbrida para o reconhecimento de entidades nomeadas em português*. 2019. Tese (Doutorado) — Universidade Federal do Espírito Santo, Vitória (Brasil), 2019.

PRIVATTO, P. I. M. *Uma abordagem para reconhecimento de entidades nomeadas usando conhecimento externo*. 2020. 88 p. Dissertação (Mestrado em Ciência da Computação) — Universidade Estadual Paulista (Unesp), Rio Claro, 2020.

SANG, E. F.; MEULDER, F. D. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*, 2003.

SANTOS, C. dos; GUIMARÃES, V. Boosting named entity recognition with neural character embeddings. Association for Computational Linguistics, Beijing, China, p. 25–33, jul 2015.

SANTOS, D. et al. Harem: An advanced NER evaluation contest for Portuguese. In: *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. Genoa, Italy: European Language Resources Association (ELRA), 2006.

SANTOS, J. et al. Assessing the impact of contextual embeddings for portuguese named entity recognition. In: IEEE. *2019 8th Brazilian Conference on Intelligent Systems (BRACIS)*. [S.l.], 2019. p. 437–442.

SANTOS, J. d. S. *Estudo comparativo de diferentes classificadores baseados em aprendizagem de máquina para o processo de Reconhecimento de Entidades Nomeadas*. 2017. 66 p. Dissertação (Mestrado em Computação Aplicada) — Universidade Estadual de Feira de Santana, Feira de Santana, 2017.

SANTY, S.; BHATTACHARYA, P. A discussion on building practical nlp leaderboards: The case of machine translation. *arXiv preprint arXiv:2106.06292*, 2021.

SILVA, A. V. *Um modelo de classificação para o Reconhecimento de Entidades Nomeadas*. 2020. 116 p. Dissertação (Mestrado em Letras) — Universidade de São Paulo, São Paulo, 2020.

SILVA, J. L. L. *Identificação de dificuldades no processo de elaboração de editais: estudo de caso na modalidade de licitação pregão eletrônico*. Rio Grande do Sul: [s.n.], 2015. 76 p.

SILVA, T. S. da. *Reconhecimento de Entidades Nomeadas em Notícias de Governo*. 2012. 99 p. Dissertação (Mestrado em Engenharia de Sistemas e Computação) — Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2012.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. Portuguese named entity recognition using bert-crf. *arXiv preprint arXiv:1909.10649*, 2019.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. BERTimbau: pretrained BERT models for Brazilian Portuguese. In: *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*. [S.l.: s.n.], 2020.

UNIÃO, B. T. d. C. da. Licitações e contratos: orientações e jurisprudência do TCU. Senado Federal, Secretaria Especial de Editoração e Publicações, 2010.

VASWANI, A. et al. Attention is all you need. In: GUYON, I. et al. (Ed.). *Advances in Neural Information Processing Systems*. [S.l.]: Curran Associates, Inc., 2017. v. 30, p. 5998–6008.

YADAV, V.; BETHARD, S. A survey on recent advances in named entity recognition from deep learning models. In: *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, 2018. p. 2145–2158.

APÊNDICES

APÊNDICE A – CLASSIFICAÇÃO DE SUBSEÇÕES TÉCNICAS OU JURÍDICAS UTILIZANDO ALGORITMOS DE APRENDIZADO DE MÁQUINA

Conforme detalhado no Capítulo 4, os editais de compras são formados por seções chamadas de **Técnicas** e **Jurídicas**. O foco deste trabalho é somente nas seções **Jurídicas**, pois são nelas que estão presentes as entidades nomeadas de interesse. Por isso, foi necessário realizar a tarefa de identificação e filtragem das seções **Técnicas**, visando manter somente as seções **Jurídicas**. Para isso, esse procedimento foi modelado como uma tarefa de classificação automática de texto cujo objetivo foi criar um modelo de Aprendizado de Máquina (AM) capaz de classificar as subseções dos editais de forma simples.

Por estar fora do escopo do trabalho, a construção do modelo não foi pautada em ajustes finos para tentar alcançar o máximo possível de desempenho, sendo que os parâmetros dos algoritmos foram utilizados, praticamente, com os valores padrões utilizados nas bibliotecas adotadas. Pelo mesmo motivo, os algoritmos de classificação testados não foram detalhados. Informações adicionais podem ser obtidas na documentação disponibilizada através dos *links* nas notas de rodapé.

Foram avaliados os seguintes classificadores: Gradiente descendente estocástico, do inglês *Stochastic Gradient Descent* (SGD), Árvore de decisão, Florestas aleatórias, Gradiente aumentado, do inglês *Gradient Boosting*, Naive Bayes multinomial (NB multinomial), Perceptron de múltiplas camadas, do inglês *Multi Layer Perceptron* (MLP), Máquina de vetores de suporte, do inglês Support Vector Machines (SVM), e XGBoost. Os parâmetros usados nos testes estão listados no Quadro 3.

Quadro 3 – Parâmetros utilizados nos modelos testados.

Modelo	**Parâmetros**
SGD[26]	$max_iter = 1000, tol = 1e-3, loss = \text{“}log\text{”}, n_jobs = -1$
Árvore de decisão[27]	$random_state = 0$
Florestas aleatórias[28]	$random_state = 0, n_jobs = -1$
Gradient Boosting[29]	$n_estimators = 100, learning_rate = 1,0, random_state = 0$
Multinomial NB[30]	$alpha = 1, fit_prior = True$
SVM[31]	$gamma = \text{“}auto\text{”}$
MLP[32]	$random_state = 1, max_iter = 300$
XGBoost[33]	$use_label_encoder = False, max_depth = 3, n_estimators = 300, learning_rate = 0,05, eval_metric = \text{“}mlogloss\text{”}$

Fonte: Elaborado pelo autor (2022).

A base de dados utilizada para treinar os modelos foi construída da seguinte maneira: **(i)** inicialmente foram utilizados 20 editais como base para realizar a etiquetagem manual das subseções, totalizando 5.276 subseções classificadas (divididas em 3.165 para treino e 2.111 para teste, na proporção 60%/40%). Vale ressaltar que esses editais não fazem parte do conjunto de editais usados para construção dos *corpora* de entidades nomeadas; **(ii)** os modelos iniciais foram treinados utilizando o *corpus* construído no passo (i); **(iii)** o modelo

de florestas aleatórias (*Random Forest*) apresentou o melhor desempenho nos testes iniciais e, portanto, foi escolhido; **(iv)** usando o melhor modelo selecionado, foram classificados outros 25 editais, dessa vez automaticamente; **(v)** os 25 editais classificados no passo (iv) foram inspecionados manualmente e quando necessário as etiquetas atribuídas pelo modelo eram corrigidas. O resultado deste processo foi a obtenção de uma nova versão do *corpus*, baseada em 45 editais, contendo 12.190 subseções classificadas (sendo 7.314 para treino e 4.876 para teste), proporcionando um acréscimo de 6.914 subseções classificadas ($\approx$ 131,05%), em comparação com a versão do *corpus* descrita no passo (i).

Todas as versões dos modelos e resultados do treino inicial foram descartados para que fosse apurado qual o modelo teria o melhor desempenho na classificação das subseções, novas versões dos modelos foram treinadas com base no *corpus* final. Os resultados obtidos neste experimento estão apresentados na Tabela 11.

Tabela 11 – Comparação dos resultados (Precisão, Cobertura e Medida-F em %) dos testes para escolha do algoritmo que foi utilizado no modelo de classificação. O maior resultado em cada medição está destacado em negrito.

Modelo	**JUR**			**TEC**		
	Precisão	**Cobertura**	**Medida-F**	**Precisão**	**Cobertura**	**Medida-F**
SGD	92,21	**99,66**	95,79	**96,67**	53,92	69,22
Árvore de Decisão	98,13	97,89	98,01	88,60	89,77	89,18
Florestas Aleatórias	**98,43**	99,03	**98,73**	94,50	**91,37**	**92,91**
Gradient Boosting	94,34	97,89	96,08	85,45	67,86	75,65
Multinomial NB	98,37	96,97	97,67	84,60	91,23	87,79
SVM	93,13	99,61	96,26	96,57	59,76	73,83
MLP	97,76	98,54	98,15	91,67	87,65	89,61
XGBoost	92,69	99,34	95,90	94,09	57,10	71,07

Fonte: Elaborado pelo autor (2022).

Com base nos resultados experimentais obtidos, o melhor modelo continuou sendo o de Florestas Aleatórias (*Random Forest*), por obter os maiores resultados com base na medida-F tanto para a classe Jurídica (**JUR**) quanto para a Técnica (**TEC**), sendo que nessa classe foi o único modelo que alcançou mais de 90% de medida-F. Além disso, ele também obteve a maior precisão para a classe **JUR** e a maior cobertura para a classe **TEC**. O segundo melhor modelo foi o SGD que atingiu maior cobertura para a classe **JUR**

[26] <https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier>
[27] <https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier>
[28] <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier>
[29] <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier>
[30] <https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB>
[31] <https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC>
[32] <https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier>
[33] <https://xgboost.readthedocs.io/en/stable/>

e maior precisão na classe **TEC**. Considerando somente a classe **JUR**, todos os modelos alcançaram bons resultados (superiores a 92%) em todas as medidas, mas observando somente a classe **TEC** os resultados foram mais diversos, principalmente na medida de cobertura (por exemplo, o modelo SGD obteve o pior resultado, com apenas 53,92%). Isso fez com que a medida-F para a classe **TEC** ficasse menor que 90% na maioria dos modelos.