UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE EDUCAÇÃO E CIÊNCIAS HUMANAS - CECH
DEPARTAMENTO DE LETRAS

LUCAS LOPES RIBEIRO

ANÁLISE DE REGRAS LINGUÍSTICAS PARA O APERFEIÇOAMENTO DE ANOTAÇÕES AUTOMÁTICAS DE *PART-OF-SPEECH*

SÃO CARLOS - SP
2023

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE EDUCAÇÃO E CIÊNCIAS HUMANAS - CECH
DEPARTAMENTO DE LETRAS

LUCAS LOPES RIBEIRO

ANÁLISE DE REGRAS LINGUÍSTICAS PARA O APERFEIÇOAMENTO DE ANOTAÇÕES AUTOMÁTICAS DE *PART-OF-SPEECH*

Trabalho de conclusão de curso apresentado ao Departamento de Letras da Universidade Federal de São Carlos, para obtenção do título de Bacharel em Linguística. Orientadora: Profª. Drª. Ariani Di Felippo

SÃO CARLOS - SP
2023

UNIVERSIDADE FEDERAL DE SÃO CARLOS
CENTRO DE EDUCAÇÃO E CIÊNCIAS HUMANAS - CECH
DEPARTAMENTO DE LETRAS

Folha de aprovação

LUCAS LOPES RIBEIRO

ANÁLISE DE REGRAS LINGUÍSTICAS PARA O APERFEIÇOAMENTO DE ANOTAÇÕES AUTOMÁTICAS DE *PART-OF-SPEECH*

Orientadora:

_____

Profª. Drª. Ariani Di Felippo
Universidade Federal de São Carlos - UFSCar

Examinador:

_____

Profª. Drª. Cláudia Dias de Barros
Instituto Federal de São Paulo (Campus de Sertãozinho) – IFSP/SRT

Agradecimentos

Gostaria de expressar meus sinceros agradecimentos às pessoas que desempenharam papéis fundamentais na realização deste trabalho de conclusão de curso. Em primeiro lugar, quero expressar minha gratidão aos meus pais, cujo apoio e incentivo ao longo de toda a minha jornada acadêmica foram essenciais para que eu pudesse alcançar este marco.

Também gostaria de estender meu agradecimento à minha orientadora, Ariani. Sua orientação perspicaz, expertise na área e paciência foram elementos-chave para a concretização deste trabalho. Suas sugestões, feedback construtivo e orientações minuciosas não apenas moldaram este projeto, mas também enriqueceram minha compreensão sobre o assunto. Sua disponibilidade para discussões e esclarecimentos foi um fator fundamental que tornou essa jornada de pesquisa mais significativa e enriquecedora.

Agradeço a todos os meus amigos e colegas que também desempenharam um papel importante, seja ao compartilhar ideias, oferecer apoio moral ou simplesmente serem ouvidos atentos durante momentos de desafio. Suas contribuições foram inestimáveis.

Obrigado.

RESUMO

A etiquetação morfossintática automática de *Part-of-speech*, também denominada PoS *tagging*, é uma tarefa essencial, pois é um dos primeiros processamentos textuais pelo qual um texto é submetido durante a análise realizada por aplicações ou métodos de Processamento Automático das Línguas Naturais. A tarefa consiste em classificar as palavras de um texto de acordo com as suas classes gramaticais. Na literatura, há diversas pesquisas voltadas para esse tipo de atividade, em sua maioria voltada para *corpus* dos gêneros mais formais como o jornalístico e acadêmico. Além disso, a *Universal Dependencies* (UD) é a teoria linguística mais adotada nas pesquisas de etiquetação automática atualmente, por apresentar diretrizes universais de etiquetas morfossintáticas. Para a língua portuguesa, ainda há poucos trabalhos baseado nesse formalismo, sobretudo quando se trata de conteúdo (principalmente, textos) gerado por usuários (CGU). Portanto, o objetivo deste trabalho foi o de analisar, refinar e avaliar um conjunto de regras de pós-edição de *tagging*, propostas por Ceregatto (2022), a partir de erros cometidos pelo modelo UDPipe 2.1 quando da anotação do *corpus* DANTEStocks, que reúne um conjunto de *tweets* do mercado financeiro. Tais regras objetivam enriquecer os métodos de *tagging* (estatísticos e/ou probabilísticos, como o UDPipe 2.1) com conhecimento linguístico para textos do tipo CGU em língua portuguesa. Como resultado, destaca-se a redução do conjunto inicial de regras, a formalização de sua descrição e a avaliação das regras refinadas que se referem à *tag* ADJ.

Palavras chaves: Etiquetação morfossintática automática, *tweet*, *Universal Dependencies*.

ABSTRACT

The automatic morphosyntactic tagging of Part-of-Speech, also known as PoS tagging, is an essential task as it is one of the initial text processing steps that a text undergoes during analysis performed by Natural Language Processing (NLP) applications or methods. The task involves classifying words in a text according to their grammatical classes. In the literature, there are numerous research efforts dedicated to this type of activity, mostly focused on corpora of more formal genres such as journalistic and academic texts. Furthermore, Universal Dependencies (UD) is the most widely adopted linguistic theory in current research on automatic tagging due to its universal guidelines for morphosyntactic labels. For the Portuguese language, there are still few works based on this formalism, especially when it comes to user-generated content (UGC). Therefore, the objective of this study was to analyze, refine, and evaluate a set of post-editing tagging rules proposed by Ceregatto (2022), based on errors made by the UDPipe 2.1 model when annotating the DANTEStocks corpus, which comprises a collection of financial market tweets. These rules aim to enrich tagging methods (statistical and/or probabilistic, such as UDPipe 2.1) with linguistic knowledge for UGC texts in the Portuguese language. As a result, the reduction of the initial set of rules, the formalization of their description, and the evaluation of refined rules referring to the ADJ tag are highlighted.

Keywords: Automatic morphosyntactic tagging, tweet, Universal Dependencies.

LISTA DE FIGURAS

LISTA DE QUADROS

SUMÁRIO

1 INTRODUÇÃO

O Processamento Automático das Línguas Naturais (PLN) é a área em que se busca dar aos computadores a habilidade de processar (interpretar e/ou gerar) língua natural em diferentes níveis linguísticos (JURAFSKY, MARTINS, 2022). No PLN, o processamento das línguas naturais pode se dar em diversas tarefas ou aplicações, como assistentes virtuais, *chatbots*, reconhecimento de fala, análise de sentimentos, tradução automática, entre outras.

Em muitas dessas aplicações, um dos primeiros processos a que um texto é submetido é a etiquetação morfossintática ou *part-of-speech* (PoS) *tagging*, que é realizada pela ferramenta denominada etiquetador (morfossintático) ou *tagger*. O processo de *tagging* consiste em identificar, a partir de um conjunto de etiquetas ou *tags* pré-definidas, a que adequadamente representa a classe gramatical de cada *token* (como palavras e sinais de pontuação) em contexto, e, na sequência, associar a etiqueta a cada *token* (JURAFSKY, MARTINS, 2022; MITKOV, 2005). O reconhecimento adequado da classe das palavras em contexto é essencial para que os sistemas de PLN possam, por exemplo, reconhecer a estrutura sintática das sentenças do texto.

No que diz respeito aos diferentes tipos de "conteúdo gerado por usuário" (CGU), como os textos produzidos em *blogs*, *Twitter*, *Facebook*, ou entre outras mídias sociais, a etiquetação morfossintática tem sido mais desafiadora, pois o alto grau de informalidade e fragmentação e a frequente ocorrência de desvios ortográficos e estruturas textuais/lexicais típicas das plataformas são fatores que dificultam a identificação automática da classe dos *tokens*.

A maior parte das pesquisas recentemente desenvolvidas sobre PoS *tagging* está relacionada ao modelo denominado *Universal Dependencies* (UD) (NIVRE, 2015; NIVRE et al, 2016 NIVRE et al., 2020) e às técnicas de redes neurais artificiais e modelos distribucionais (SANGUINETTI *et al.,* 2020). A UD é um modelo gramatical que fornece um esquema "universal" para representar a morfologia e a sintaxe das línguas naturais. Sendo a evolução do esforço de uma ampla comunidade colaborativa de trabalho, esse modelo tem sido usado como referência para a construção de *treebanks,* ou seja, *corpora* (morfo-)sintaticamente anotados). Entre esses *treebanks,* encontram-se vários *tweebanks,* que são *corpora* compostos exclusivamente por postagens compiladas do *Twitter*. A proeminência dos *tweebanks* anotados com UD no cenário do PLN se deve, sobretudo, à adaptabilidade do modelo às particularidades dos *tweets* em todos os níveis de representação.

Com base na literatura internacional, Silva (2022) desenvolveu as primeiras investigações sobre o processo de PoS *tagging* para CGU em português a partir de *corpus* de *tweets* anotados com base no modelo UD. Especificamente, o autor customizou os métodos de

tagging do estado-da-arte para *corpus* de *tweets* em português. O *corpus* utilizado pelo autor foi o chamado DANTEStocks, que é formado por *tweets* do mercado financeiro e está sendo anotado segundo o modelo UD (DI FELIPPO *et al*., 2021). Utilizando esse *corpus* para treinamento e teste dos métodos de PoS, Silva (2022) verificou que o método de melhor performance para a tarefa em questão foi o UDPipe 2.1, o qual atingiu 95% de f-*score.*

Ceregatto (2022), motivado pela relevância do *corpus* DANTEStocks e pelas pesquisas incipientes sobre *tagging* para CGU em português, realizou, além de uma caracterização do *corpus*, uma análise dos erros cometidos pelo UDPipe 2.1 e propôs regras de pós-edição para futura correção dos erros. Especificamente, ele verificou que, do total de etiquetas atribuídas pelo modelo, 4,7% delas foram identificadas erroneamente e propôs um conjunto preliminar de 93 regras de pós-edição, as quais, no entanto, não foram avaliadas.

Diante desse cenário, fez-se, neste trabalho, uma análise das regras propostas por Ceregatto (2022) com o objetivo de refiná-las e, por fim, testá-las para avaliação.

Para tanto, este trabalho foi equacionado em 5 Seções. Na Seção 2, apresenta-se uma breve revisão literária sobre *tagging*, incluindo as pesquisas recentes baseadas no modelo UD e as especificamente focadas no processamento de UGCs. Na Seção 3, apresenta-se uma análise crítica das regras de Ceregatto e propostas de refinamento. Na Seção 4, apresentam-se a avaliação das regras e discussão dos resultados. Por fim, na Seção 5, apresentam-se as considerações finais deste trabalho, enfatizando contribuições, limitações e pesquisas futuras.

2 REVISÃO DA LITERATURA

2.1 Etiquetação morfossintática automática e os CGUs

Para identificar a classe dos *tokens*, que é o objetivo da tarefa de PoS *tagging*, o texto de entrada precisa passar pela etapa de pré-processamento denominada *tokenização*, que consiste em identificar os *tokens* que compõem as sentenças, os quais podem ser palavras, pontuação e símbolos ou caracteres especiais. A depender da base teórica ou da aplicação de PLN, as unidades a serem *tokenizadas* podem variar. Dessa forma, como apontado por Ceregatto, um nome próprio como “Nova York”, por exemplo, pode ser tratado como um *token* individual ou, por outro lado, desmembrado em dois *tokens*. As contrações em língua portuguesa, como “do”, por sua vez, tendem sempre a ser decompostas em dois *tokens*, isto é, “de” + “o”.

Na literatura, há vários métodos de *tagging* desenvolvidos segundo diferentes paradigmas, como métodos baseados em regras (KLEIN; SIMMONS, 1963), métodos probabilísticos (KEPLER, 2010), conexionistas (BOHNET et al., 2018), entre outros.

Com a crescente relevância das redes sociais, o processamento automático dos textos produzidos diariamente pelos usuários dessas redes tem se tornado objeto de interesse para o PLN. E esse interesse provém principalmente do objetivo de desenvolver aplicações como a análise de sentimento, que consiste em extrair informações sobre os sentimentos presentes nos textos, podendo ser utilizado para avaliação de um produto ou serviço.

Os desafios para o processamento desse material textual são muitos, sendo resultantes sobretudo da (i) informalidade, (ii) falta de padrão ortográfico e de (iii) elementos típicos das plataformas. Para o inglês, há diversas pesquisas voltadas para CGU (p.ex.: LYNN et al. 2015; BOSCO et al. 2016; PROISL 2018; REHBEIN et al. 2018; BEHZAD, ZELDES 2020).

No que diz respeito à língua portuguesa, tais pesquisas ainda são incipientes. Silva (2022) investigou pioneiramente a tarefa de *tagging* baseada no modelo UD para *tweets* em português. Especificamente, o objetivo do autor foi o de investigar alguns métodos do estado-da-arte, como o UPipe 1 (STRAKA; HAJIČ; STRAKOVÁ, 2016), o UDPipe 2.1 (STRAKA; STRAKOVÁ; HAJIC, 2019) e o Udify (KONDRATYUK; STRAKA, 2019) para o cenário em questão. Tais métodos se caracterizam por utilizar representações contextuais baseadas no BERT (DEVLIN et al., 2019) e em rede neural. Para otimizar esses métodos para os *tweets,* os autores utilizaram o DANTEStocks, que, como mencionado, contém *tweets* do mercado de financeiro e está sendo anotado segundo o modelo UD (DI FELIPPO *et al*., 2021). Com base nesse *tweebank* específico, Silva (2022) treinou vários métodos, sendo que o de melhor performance foi o UDPipe 2.1, que atingiu 95% de f-*score* (ou medida-f).

2.2 O *corpus* DANTEStocks

O DANTEStocks é um *corpus* de CGU em português composto por *tweets* sobre o mercado financeiro. Ele resultou do refinamento e da anotação morfossintática do *corpus* de Silva *et al*. (2020), cuja compilação se baseou na ocorrência de menos um *ticker*[1] de uma das 73 ações do IBovespa. Atualmente, esse *corpus* possui 4.048 *tweets*, que englobam aproximadamente 81 mil *tokens*. Quanto pré-processamento, o *corpus* não foi submetido a nenhuma normalização (isto é, conversão da linguagem em um formato "padrão" ou mais formal) e, por ter sido compilado em 2014, os *tweets* têm no máximo 140 caracteres. Ademais, os *tweets* apresentam diferentes constituições internas, podendo apresentar (i) uma ou mais sentenças bem delimitadas (1) e (2), ausência de pontuação (3) ou pontuação equivocada (4), (iii) fragmentação (5), e (iv) colagens de manchetes de outras fontes (6) (DI-FELIPPO *et al.,* 2021).

[1] Combinação composta por quatro letras e um número que se refere tanto ao nome da empresa quanto ao tipo de ação, como "petr4" em (1).

(1) Sera k petr4 já entrou na baixa?
(2) PETR4 subiu na bolsa 13,50. Muito bem, surpreso com o resultado.
(3) #PETR4 #PETROBRAS a R$13,13. Pronto! O #PT conseguiu fazer propaganda eleitoral antecipada O que a @user tem a dizer sobre isso?
(4) Bom dia Marcos, Alguma previsão para petr4?!
(5) #GGBR4 Suportes e resistências http://t.co/Azw6yIEVI9
(6) Logística, ex-LLX, anuncia prejuízo de R$ 135,8 milhões em 2013: A Prumo Logística, ex-LLX (LLXL3), divu... http://t.co/LwmlKPqssk.

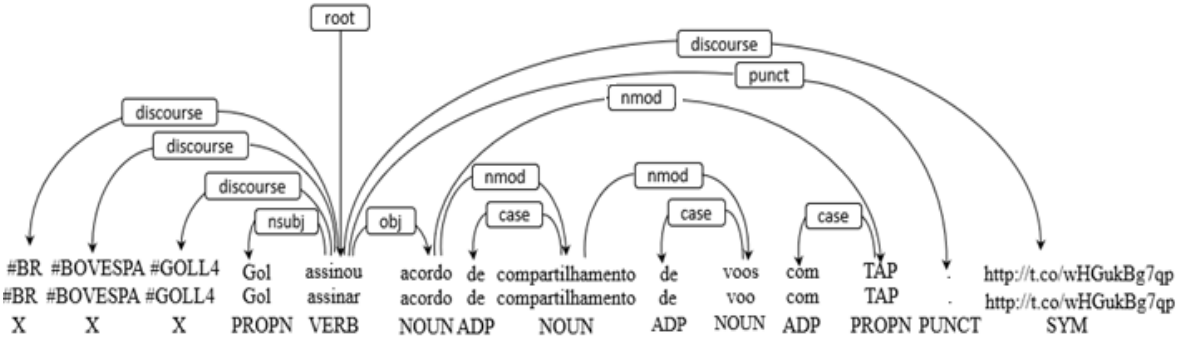
Quanto à *tokenização*, destaca-se que, por seguir a UD, que é um modelo gramatical lexicalista, as unidades básicas de anotação do DANTEStocks são as palavras sintáticas[2] (ou *tokens*). A partir de uma caracterização do *corpus* na qual fenômenos gráficos e lexicais foram sistematizados, diretrizes de segmentação lexical foram definidas. Um desses fenômenos são as contrações, isto é, formas abreviadas de duas palavras funcionais com remoção de espaços e/ou letras. Exemplos de contrações são "oq" e "pq", sendo a primeira constituída por dois pronomes ("o"+"que") e a segunda por palavras de categorias diferentes ("por"+"que") (preposição e pronome, respectivamente). Para ambos os casos, tem-se como diretriz de *tokenização* a decomposição em dois *tokens*. Essa diretriz foi definida para que cada um dos elementos constitutivos de uma contração pudesse ser anotado adequadamente segundo o modelo UD[3]; por exemplo, para que cada elemento pudesse ser associado a seu respectivo lema ou forma canônica (DI FELIPPO et al., 2021).

Quanto à anotação, a UD prevê 2 níveis. No nível morfológico, especificam-se 3 informações: lema, etiqueta morfossintática e traços lexicais/gramaticais (*features*). No nível sintático, a anotação se dá por relações de dependência (*deprels*), que são binárias e assimétricas. A representação básica de uma estrutura de dependências é arbórea, na qual uma palavra é o *root* (raiz) da sentença. Na Figura 1, ilustra-se a anotação-UD completa de um *tweet* do *corpus* com base em Sanguinetti et al. (2022). Na Figura 1, as etiquetas de PoS estão em caixa alta, como NOUN para "acordo". Acima, estão os lemas, como "voo" para "voos". As *deprels* estão indicadas por setas rotuladas que se originam no *head* e se destinam ao dependente. Na figura, "acordo" é dependente de "assinou" e estes estão conectados pela *deprel* `obj` (objeto direto). O verbo "assinou" é o `root` dessa representação. Os traços não constam na

[2] Palavra sintática (*syntactic word*) é a unidade mínima a que corresponde uma função sintática.
[3] Mais informações sobre os fenômenos lexicais e as diretrizes de *tokenização* podem ser encontrados em Di Felippo *et al.* (2021).

Figura 1, mas “acordo”, por exemplo, tem os traços-valores: `Gender=Masc` e `Number=Sing`. Atualmente, o DANTEStocks possui anotação de nível morfológico, a qual é a seguir.

Figura 1. Exemplo de *tweet* do DANTEStocks com anotação-UD.



Fonte: Elaborada pelo autor

2.3 Anotação de *PoS* do *corpus* DANTEStocks

A anotação de PoS dos *tweets* foi feita de forma semiautomática, isto é, as postagens foram anotadas automaticamente e revisadas manualmente na sequência. A *tokenização* foi feita pelo *tokenizador* simbólico de *tweets* do pacote NLTK[4], o qual foi enriquecido por regras específicas para a identificação dos fenômenos do DANTEStocks. Tais regras foram elaboradas com base na caracterização do estatuto de *token* dos fenômenos lexicais de Di Felippo *et al.* (2021).

Para a anotação de PoS, os 4.048 *tweets* foram divididos em 13 pacotes. Com exceção do 1º pacote, composto por 147 *tweets*, os outros 12 continham cerca de 325 *tweets* cada. Essa divisão foi feita para que a anotação semiautomática pudesse ser feita de modo incremental. A anotação semiautomática em questão envolveu o treinamento do *parser* UDPipe 2.1[5] (STRAKA, 2018) no DANTEStocks, uma vez que esse *parser* havia sido treinado apenas a partir de textos com linguagem padrão (notícias jornalísticas) em português. Especificamente, após a revisão manual da anotação automática do pacote 0, os dados do pacote foram submetidos ao UDPipe 2.1 para treinamento, de forma a preparar o *parser* para anotar os *tweets* do pacote 1. Após a revisão manual da anotação de PoS do pacote 1, os *tweets* dos pacotes 0 e 1 foram submetidos ao UDPipe 2.1 e assim sucessivamente até que todos os 8 pacotes (0-7) tivessem sido anotados e revisados.

Vale ressaltar que o UDPipe 2.1 empregou o conjunto de 17 *tags* de PoS. Vê-se no Quadro 1 que as *tags* são divididos em: (i) palavras da classe aberta, as quais podem ser constituídas por um número infinito de palavras, pois recebem novas palavras, (ii) palavras da

[4] Trata-se do NLTK *TweetTokenizer* (https://www.nltk.org/api/nltk.tokenize.html)
[5] https://ufal.mff.cuni.cz/udpipe/2.

classe fechada, as quais englobam um conjunto finito de palavras e raramente ocorre a adição de uma nova, e a (iii) categoria "outros", que engloba sinais de pontuação, símbolos e qualquer unidade linguística que não pertença às demais classes.

Quadro 1: Conjunto de etiquetas morfossintáticas do modelo UD (versão 2.1).

Palavras de classe aberta	Palavras de classe fechada	Outros
ADJ	ADP	PUNCT
ADV	AUX	SYM
INTJ	CCONJ	X
NOUN	DET	
PROPN	NUM	
VERB	PART	
	PRON	
	SCONJ	

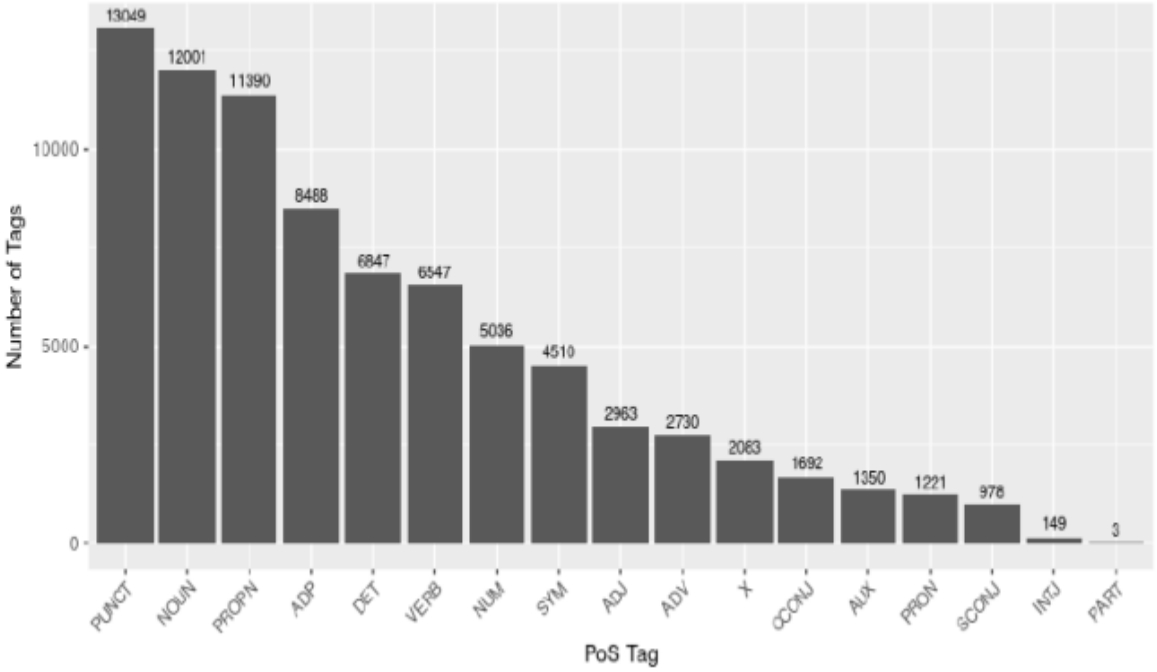
Fonte: https://universaldependencies.org/u/pos/index.html

A revisão manual das *tags* de PoS foi guiada pelo manual de Duran (2022), que contém diretrizes para a anotação UD do português, e pelo manual de Di-Felippo *et al.* (2022), que engloba diretrizes específicas para a anotação de PoS dos fenômenos típicos dos *tweets* do mercado financeiro. Assim, cada um dos 13 pacotes de *tweets* foi submetido à revisão de três anotadores humanos diferentes e somente os casos de divergência entre eles foram adjudicados por uma linguista sênior. A revisão foi feita por meio da ferramenta *online* Arborator-Grew (GUIBON *et al.*, 2020), desenvolvida para as tarefas de anotação/revisão de *corpora* anotados segundo a UD. Especificamente, utilizou-se a versão refinada por Miranda e Pardo (2022a)[6]. A anotação de PoS manualmente revisada é considerada "referência" e foi esta utilizada por Silva (2022) para a exploração dos métodos de *tagging* para *tweets* em português.

Antes de se focar na análise feita por Ceregatto (2022) a respeito dos erros cometidos pelo UDPipe 2.1, destaca-se a distribuição das *tags* no *corpus* (Figura 2). Observa-se que as etiquetas mais utilizadas foram PUNCT, NOUN e PROPN. Ceregatto interpreta que: (i) a alta frequência de PUNCT se deve pela diretriz de segmentar cada sinal em uma sequência de repetições (como "!!!!"), resultando, nesse caso, em 4 *tokens* individuais ("!"+"!"+"!"+"!"); (ii) a frequência alta de NOUN evidencia a natureza nominal dos *tweets*, os quais são composto majoritariamente por sintagmas nominais, e (iii) a ocorrência frequente de PROPN se deve não só ao domínio, repleto de nomes de empresas e pessoas, mas também pelo fato de que os *tickers*, usados para compilar os *tweets*, ocorre em todos os *tweets* e foram anotados como PROPN.

[6] https://arborator.icmc.usp.br/#/

Figura 2: Distribuição das tags *PoS* no DANTEStock.



Fonte: Ceregatto (2022).

2.4 Análise dos erros de *tagging* no DANTEStocks

Para propor regras de correção a serem aplicadas após o processo de *tagging*, Ceregatto realizou uma análise dos erros cometidos pelo UDPipe 2.1. Vale ressaltar que, quando Silva (2022) explorou os métodos do estado-da-arte de *tagging* para os *tweets* em português, o *corpus* DANTEStocks ainda não havia sido refinado e, por isso, continha os 4,517 *tweets* originalmente compilados por Silva *et al.* (2020). Ademais, havia apenas 8 (0-7) pacotes com anotação de referência. Dessa forma, os métodos foram investigados com apenas 8 pacotes, totalizando 2,737 *tweets*. Isso quer dizer que a análise dos erros realizada por Ceregatto (2022) é parcial.

O Quadro 2 dispõe a medida-f obtida pelo UDPipe 2.1 para cada *tag*. Nele, as *tags* PoS foram ordenadas de maneira decrescente em função da referida medida. Assim, tem-se que a *tag* de medida-f mais alta ocupa o topo da lista e a *tag* PoS de medida-f mais baixa ocupa a última colocação da lista. O Quadro 3 apresenta os números absolutos dos erros e acertos em função de cada *tag*. Observa-se no Quadro 2 que os resultados variam entre 99,33% e 44,44%, sendo que as etiquetas com melhor e pior desempenho são, respectivamente, CCONJ e INTJ. De acordo com Ceregatto, uma possível razão para essa diferença seria a diferença na frequência de ocorrências dessas *tags* no *corpus*, uma vez que INTJ é a segunda menos frequente (149 casos), ao passo que CCONJ, mesmo não estando entre as mais frequentes no *corpus*, possui frequência simples bem mais alta que INTJ (1642 ocorrências) (cf. Figura 2).

Quadro 2: Medida-F das *tags* obtidas pelo UDPipe 2.1.

Tag	UDPipe 2.1	Qt.
CCONJ	99.33	1692
PUNCT	99.30	13049
SYM	99.13	4510
NUM	97.99	5036
DET	97.70	6847
ADP	97.45	8488
PROPN	95.77	11390
VERB	94.88	6547
ADV	94.69	2730
NOUN	94.20	12001
AUX	92.39	1350
PRON	92.10	1221
ADJ	90.46	2963
SCONJ	89.90	976
X	79.06	2083
INTJ	44.44	149

Fonte: Silva (2022), apud Ceregatto (2022)

Quadro 3: Quantidade de erros e acertos por *tag*.

tag PoS	Qt total de casos	Qt de acertos	Qt de erros
ADJ	343	313	30
ADP	977	956	21
ADV	379	357	22
AUX	188	176	12
CCONJ	223	221	2
DET	862	848	14
INTJ	24	8	16
NOUN	1341	1276	65
NUM	513	512	1
PRON	217	204	13
PROPN	1367	1324	43
PUNCT	1283	1278	5
SCONJ	145	138	7
SYM	346	342	4
VERB	890	852	38
X	656	489	167
TOTAL	**9762**	**9311**	**468**

Fonte: Silva (2022), apud Ceregatto (2022)

Para a análise dos erros, Ceregatto refinou o *corpus* de 2,737 *tweets* empregados na exploração dos métodos de *tagging*, excluindo (i) *tweets* duplicados (sobretudo em decorrência da função *retweet* da plataforma), (ii) *tweets* contendo erros na anotação de referência, (iii) *tweets* com erros de digitação no co-texto e (iv) *tweets* com problemas de *tokenização*.

Além disso, o autor desconsiderou os erros referentes às etiquetas PROPN, AUX e X, pois a revisão manual destas ainda estava sendo realizada em função da definição de novas diretrizes de anotação-UD para o português. Assim, os erros totais passaram de 468 para 128, representando uma diminuição de 72% dos casos.

No Quadro 4, reúnem-se os 128 casos de erros organizados em função da *tag* correta que deveria ter sido anotada pelo UDPipe 2.1 (coluna "*Tag*-alvo"). O Quadro 4 também evidencia a *tag* erroneamente identificada pelo *parser* (coluna "Confusão"). Como exemplo, observa-se que há 19 casos no Quadro 4 em que o *token* deveria ter sido automaticamente anotado como ADJ, mas o *parser* erroneamente etiquetou 1 desses casos como ADV, 10 casos como NOUN e outros 8 casos como VERB.

Quadro 4: Quantidade de erros após refinamento (continua)

***Tag*-alvo**	**Confusão**	**Quantidade**	**Total por *tag***
ADJ	ADV	1	19
	NOUN	10	
	VERB	8	
ADP	ADV	1	14
	DET	1	
	NOUN	2	
	SCONJ	10	
ADV	ADJ	3	18
	ADP	2	
	NOUN	7	
	PRON	2	
	SCONJ	3	
	VERB	1	
DET	ADJ	2	14
	ADP	4	
	NUM	2	
	PRON	4	
	SCONJ	2	
INJT	ADJ	2	8
	ADV	3	
	NOUN	2	
	PRON	1	

NOUN	ADJ	12	23
	NUM	4	
	PRON	1	
	SYM	1	
	VERB	5	
PRON	ADV	2	8
	DET	3	
	NOUN	2	
	SCONJ	1	
SCONJ	ADP	1	7
	ADV	3	
	PRON	3	
SYM	NOUN	1	3
	PUNCT	2	
VERB	ADJ	3	13
	ADP	2	
	NOUN	8	
TOTAL		**128**	**128**

Fonte: Ceregatto (2022).

A partir dos 128 casos, Ceregatto propôs um conjunto de 93 regras para serem empregadas após o processo de *tagging*, como uma estratégia de pós-edição. Tais regras, em particular, foram propostas especificamente para corrigir erros de etiquetação automática de *tokens* pertencentes às classes ADJ, ADP, ADV, DET, INTJ, NOUN, PRON, SCONJ, SYM e VERB. O conjunto total de 93 regras está descrito no Apêndice 1 deste trabalho.

Para ilustrar, o Quadro 5 exibe o conjunto de 15 regras de pós-edição para os casos de ADJ erroneamente etiquetados pelo *parser*. Esse Quadro, que ilustra a descrição das regras propostas por Ceregatto (2022), contém 7 colunas. Nas 3 primeiras, estão, respectivamente, o número da regra, a etiqueta errada e a etiqueta de PoS correta. Na 4ª coluna, exibe-se a condição necessária para que a substituição prevista na 5ª coluna possa ser aplicada. A 6ª coluna apresenta um caso de etiquetação problemática do *corpus* que originou a regra em questão e exemplifica a substituição prevista na 5ª coluna. No caso da primeira regra, por exemplo, deve-se lê-la da seguinte forma: “se a palavra “certo” foi etiquetada como ADV, substituir a *tag* PoS ADV para ADJ.”

Quadro 5: Regras de pós-edição para os erros referentes a tag ADJ.

Qt	DE	PARA	Condição	Regra	Exemplo	Casos
1	ADV	ADJ	Se token="certo"	Substituir ADV por ADJ	certo/ADJ>ADV	1
2	NOUN	ADJ	Se o token for precedido por DET e NOUN	Substituir por NOUN por ADJ	em/ADP a/DET sexta/NOUN **passada**/ADJ>NOUN	2
3	NOUN	ADJ	Se o token for imediatamente precedido por DET e sucedido por NOUN.	Substituir NOUN por ADJ	a/DET **corrente**/NOUN>ADJ gravidade/NOUN	1
4	NOUN	ADJ	Se o token for imediatamente precedido por ADP e sucedido por NOUN.	Substituir NOUN por ADJ	de/ADP **módico**/NOUN>ADJ repique/NOUN	1
5	NOUN	ADJ	Se o token anterior for NOUN, substituir tag do token "log" por ADJ.	Substituir NOUN por ADJ	escala/NOUN **log**/ADJ>NOUN	1
6	NOUN	ADJ	Se token="casado"	Substituir NOUN por ADJ	**casado/ADJ>NOUN**	1
7	NOUN	ADJ	Se token="coitada", primeiro token da sentença e seguido por ADP	Substituir NOUN por ADJ	**coitada/ADJ>NOUN**	1
8	NOUN	ADJ	Se o token for seguido por PUNCT (/) + ADJ	Substituir NOUN por ADJ	**médio/ADJ>NOUN /PUNCT longo/NOUN**	1
9	NOUN	ADJ	Se token="melho"	Substituir qualquer tag PoS por ADJ	**melho/NOUN>ADJ**	1
10	NOUN	ADJ	Se token="voláteis"	Substituir qualquer tag PoS por ADJ	**voláteis/NOUN>ADJ**	1
11	VERB	ADJ	Se token 1E e 2E = ADJ + NOUN	Substituir VERB por ADJ	lucro/NOUN líquido/ADJ **consolidado**/ADJ>VERB	2
12	VERB	ADJ	Se token 1E="trade"	Substituir VERB por ADJ	Trade/NOUN **fechado**/ADJ>VERB	1
13	VERB	ADJ	Se token="lindo"	Substituir qualquer tag PoS por ADJ	lindo/ADJ>VERB	1
14	VERB	ADJ	Se token 1E="risco"	Substituir VERB por ADJ	risco/NOUN **elevado**/ADJ>VERB	1
15	VERB	ADJ	Se tokens 1D, 2D e 3D = ADP + DET + NOUN/PROPN	Substituir VERB por ADJ	para/ADP o/DET bebê/NOUN **abandonado**/ADJ>VERB	3

Fonte: Ceregatto (2022).

A seguir, apresenta-se uma análise crítica das 93 regras do autor, destacando vantagens e desvantagens, e apresentando propostas de refinamento.

3. ANÁLISE CRÍTICA DAS REGRAS

3. 1 Análise geral

O uso de regras heurísticas para melhorar a precisão de etiquetadores de PoS não é uma técnica de PLN recente. No entanto, as contribuições de Ceregatto (2022) são relevantes no que tange à etiquetação de CGU, particularmente de *tweets*.

Ao analisar as 93 regras, as quais seguem o formato lógico (*se, então*) como exemplificado no Quadro 5, constatou-se que 61 delas são lexicalizadas, pois suas condições de aplicação (isto é, informação prevista na 4ª coluna) englobam a ocorrência de uma ou mais unidades lexicais ou *tokens*, e 32 regras são classificadas como não-lexicalizadas, uma vez que suas condições envolvem a ocorrência de *tags* PoS e não de *tokens* ou palavras específicas. O

fato de a maior parte das regras ser lexicalizada está relacionado à ocorrência de apenas um caso de etiquetação equivocada, o qual, aliás, originou a regra.

Além disso, por se tratar de *tweets* de um domínio especializado, como o do mercado financeiro, as regras foram organizadas em dois grupos. Em um deles, tem-se as regras ditas de "língua geral" e, no outro, as regras relativas ao "domínio". As do primeiro grupo são aquelas que correspondem a palavras do léxico da língua portuguesa geral, e as do segundo grupo dizem respeito a *tokens* ou fenômenos oriundos do léxico do mercado financeiro ou do próprio *Twitter*. No caso, há 78 regras de "língua geral" e 15 de "domínio".

Diante disso, pode-se dizer que a maior parte das regras foi proposta para o tratamento de erros pontuais ligados à língua geral, tendo em vista que elas são lexicalizadas e de língua geral. Para ilustrar, tem-se a regra lexicalizada 9, a qual prevê que, se o *token* "melho" (resultado do truncamento[7] de "melhor") tiver sido etiquetado como NOUN, deve-se substituir a *tag* PoS NOUN por ADJ. Dessa forma, pode-se dizer que várias dessas regras não possuem uma capacidade de generalização.

Quanto às regras não-lexicalizadas, ressalta-se que elas demonstram ter uma complexidade maior, pois se baseiam na ocorrência de uma classe de palavra (ou *tag*) ou mesmo de uma sequência de classes ou *tags* e não apenas de uma palavra ou *token* específico. E, por essa mesma razão, pode-se dizer que elas têm uma capacidade maior de generalização, podendo, assim, solucionar uma quantidade maior de desvios, tanto no *corpus* de análise, quanto em diferentes textos.

No entanto, por se tratar de regras de pós-edição com o objetivo exclusivo de corrigir os desvios cometidos pelo modelo UDPipe 2.1 no *corpus* DANTEStocks, a análise restringiu-se apenas ao potencial de correção que elas possuem.

Tendo isso em vista, na seção seguinte, exemplifica-se a análise detalhada das 93 regras por meio da análise das regras referentes à *tag* ADJ. Em outras palavras, ilustra-se a análise crítica do trabalho de Ceregatto com base nas 15 regras contidas no Quadro 5.

3. 1 Exemplo de análise detalhada das regras

Especificamente, as 15 regras em questão foram propostas para que o UDPipe 2.1 pudesse corrigir as confusões ocorridas entre a *tag* ADJ, por um lado (etiqueta correta) e as *tags* ADV,

[7] Truncamento é um dos fenômenos linguísticos tipicamente encontrados em textos do tipo CGU, como os *tweets*. Esse fenômeno, nos *tweets*, resulta especificamente do limite de caracteres imposto pela plataforma que, no caso do DANTEStocks, é de 140 caracteres (cf. DI-FELIPPO et al., 2021).

NOUN e VERB, por outro lado. Portanto, todas as regras visam alterar a etiqueta atribuída inicialmente pelo *parser* (ADV, NOUN ou VERB) para ADJ.

3. 1. 1 Análise das regras: os casos de ADJ etiquetados como ADV, NOUN ou VERB

Regra 1. Diante da ocorrência do *token* "certo" etiquetado como ADV, essa regra prevê a alteração da *tag* ADV para ADJ. Trata-se de uma regra lexicalizada e simples. Ela foi elaborada para resolver apenas a etiquetação da palavra "certo", que, nos pacotes de treinamento (0-7) utilizados pelo *parser*, havia ocorrido apenas como ADJ. Isso quer dizer que, diante dos dados até então anotados do *corpus* DANTEStocks, a regra parece pertinente. No entanto, se houver casos em que "certo" seja de fato um ADV ou NOUN no restante do *corpus*, por exemplo, essa regra poderá gerar problemas. Dessa forma, não é possível garantir que todas as ocorrências dessa palavra quando etiquetada como ADV se trata de ADJ.

Regra 2. Diante da ocorrência de um *token* etiquetado pelo *parser* como NOUN, essa regra prevê que a etiqueta seja alterada para ADJ caso o *token* em questão seja precedido, na primeira posição à esquerda, por DET e, na segunda posição à esquerda, por NOUN. Trata-se de uma regra não-lexicalizada e que se baseia na ocorrência de um padrão linguístico (isto é, sequência de etiquetas) como condição, tendo em vista que a sequência DET + NOUN + ADJ é mais frequente do que DET + NOUN + NOUN. Em outras palavras, essa regra se pauta no fato de que é mais provável a ocorrência de um ADJ após a sequência DET + NOUN do que de um outro NOUN. Sendo assim, essa regra parece ter potencial de aplicação.

Regra 3: Essa regra também busca corrigir um *token* etiquetado como NOUN para ADJ. Para tanto, é preciso que o *token* anterior (à esquerda do *token*-alvo) seja DET e o posterior (à direita) seja NOUN. Assim, diante de uma sequência como DET + NOUN + NOUN, essa regra define que o primeiro NOUN seja substituído por ADJ. Vê-se, aqui, que há um conflito com a Regra 2. Para resolvê-lo, pode-se, por exemplo, verificar o impacto de cada uma das regras conflitantes no desempenho da etiquetação automática e, com isso, implementar apenas a de melhor performance. De acordo com as descrições de Ceregatto, a Regra 2 tem potencial de resolver dois conflitos encontrados durante suas análises, enquanto a Regra 3 corrigiria apenas um desvio. Visto isso, pode-se decidir previamente manter a Regra 2, embora ainda seja necessário realizar uma análise mais aprofundada. Outra possibilidade seria a de refinar as Regras 2 e 3 inserindo mais condições contextuais.

Regra 4. Diante da ocorrência de um *token* etiquetado pelo *parser* como NOUN, essa regra prevê que a etiqueta seja alterada para ADJ caso o *token* em questão seja precedido por ADP e sucedido por NOUN. Essa regra pauta-se no fato de que a sequência de *tags* ADP + NOUN + NOUN é menos frequente em português do que ADP + ADJ + NOUN. Essa regra parece não causar nenhum conflito com as demais regras.

Regra 5. Diante da ocorrência do *token* "log" etiquetado automaticamente como NOUN, essa regra prevê que a etiqueta seja alterada para ADJ caso o *token* em questão seja precedido por NOUN. Trata-se de uma regra lexicalizada e que envolve conhecimento do domínio do mercado financeiro. Diz-se isso porque, no DANTEStocks, "log" é a redução do adjetivo "logarítmico", ocorrendo na sequência de um NOUN (como em "escala log"). Portanto, essa regra parece ser pertinente para a correção do erro.

Regra 6. Diante da ocorrência do *token* "casado" etiquetado como NOUN, essa regra prevê que a etiqueta seja alterada para ADJ. Trata-se de uma regra lexicalizada, simples e determinista. Portanto, com base nela, todas as ocorrências do *token* "casado" como NOUN serão alteradas para ADJ. Portanto, essa regra parece ser pertinente para a correção do erro específico.

Regra 7. Diante da ocorrência do *token* "coitada" etiquetado automaticamente como NOUN, essa regra prevê que a etiqueta seja alterada para ADJ caso o *token* em questão esteja no início do *tweet* e seja sucedido por ADP. É uma regra simples e possui potencial para melhorar o desempenho do modelo.

Regra 8. Diante da ocorrência de um *token* etiquetado automaticamente como NOUN, essa regra prevê que a etiqueta seja alterada para ADJ caso o *token* em questão seja seguido de PUNCT (na primeira posição à direita) e ADJ (na segunda posição à direita). Isso porque a barra ("/"), nesse caso, funciona como uma conjunção, ligando, assim, dois *tokens* de mesma classe ou *tag* (ADJ + PUNCT + ADJ) (como em "médio/longo") e não de classes distintas (NOUN + PUNCT + ADJ). É uma regra coerente e que não causa conflitos com as demais, sendo possível corrigir diversos casos do tipo.

Regra 9. Diante da ocorrência do *token* "melho" etiquetado automaticamente com qualquer uma das 17 *tags* da UD, essa regra prevê que a etiqueta seja alterada para ADJ. Trata-se de uma regra lexicalizada e que busca corrigir um erro de *tagging* causado pelo truncamento específico da palavra "melhor". Por essa razão, essa regra é pontual.

Regra 10. Diante da ocorrência do *token* “voláteis” etiquetado automaticamente com qualquer uma das 17 *tags* da UD, essa regra prevê que a etiqueta seja alterada para ADJ. Trata-se de uma regra lexicalizada e que busca corrigir especificamente a etiquetação da palavra “voláteis”. Assim como as regras 1, 6 e 7, a regra 10 é simples, determinista e pontual.

Regra 11. Diante da ocorrência de um *token* etiquetado automaticamente como VERB, essa regra prevê que a etiqueta seja alterada para ADJ caso o primeiro *token* à esquerda seja um ADJ e o segundo à esquerda seja um NOUN (com em “lucro líquido consolidado”). Acredita-se que a regra possui potencial para corrigir o erro em questão e, com isso, contribuir para uma melhor etiquetação automática.

Regra 12. Diante da ocorrência de um *token* etiquetado automaticamente como VERB, essa regra prevê que a etiqueta seja alterada para ADJ caso o primeiro *token* à esquerda seja “trade/NOUN”. Essa regra, embora não especifique isso, parece ser uma regra específica para a expressão “trade fechado”, em que se tem NOUN + ADJ no DANTEStocks. Da forma como está descrita, essa regra pode gerar problemas, uma vez que levaria qualquer verbo após “trade” (como em “o trade fechou...”) a ser etiquetado como ADJ. Ademais, diz-se que a regra parece ser relativa à expressão “trade fechado” também porque ela foi elaborada apenas com base em um único caso de erro, de acordo com a descrição de Ceregatto.

Regra 13. Diante da ocorrência do *token* “lindo” etiquetado automaticamente como VERB, essa regra prevê que a etiqueta seja alterada para ADJ. É uma regra coerente, tendo em vista que não existe a possibilidade de “lindo” ser etiquetado corretamente como VERB.

Regra 14. Diante da ocorrência de um *token* etiquetado pelo *parser* como VERB, essa regra prevê que a etiqueta seja alterada para ADJ quando o *token* precedente for “risco”. Assim como a regra 12, essa também possui problemas na sua descrição, pois não fica claro se se trata de qualquer *token* ou de “elevado” apenas, como indicado no exemplo. Assim, entende-se que a regra se refere exclusivamente à expressão “risco elevado”.

Regra 15. Diante da ocorrência de um *token* etiquetado automaticamente como VERB, essa regra prevê que a etiqueta seja alterada para ADJ caso o primeiro *token* à esquerda seja NOUN/PROPN, o segundo à esquerda seja DET e o terceiro seja ADP, evidenciando o padrão ou sequência ADP+DET+NOUN/PROPN+ADJ, o qual é comum quando se observa particípios, como é o caso do exemplo (“para o bebê abandonado”). Na descrição, no entanto,

Ceregatto se equivocou, pois definiu as *tags* contextuais da condição como sendo à direita do *token* em questão. Ademais, cabe ressaltar que a condição dessa regra pode ser simplificada, considerando-se apenas os dois *tokens* imediatamente à esquerda, ou seja, DET + NOUN/PROPN, assim como a Regra 2, pois tratam do mesmo fenômeno linguístico.

Tendo em vista todas as colocações aqui apresentadas, discutem-se, na seção seguinte, algumas ações para refinar essas regras.

3.3 Proposta de aprimoramento das regras

Diante da análise realizada anteriormente, definiram-se neste trabalho algumas estratégias para aprimorar as regras de Ceregatto (2022).

O primeiro aprimoramento resulta de se verificar a existência de algumas variações e inconsistências na forma em que as regras são descritas. Assim, o primeiro aprimoramento buscou padronizar as regras. Para tanto, adotou-se a metodologia trifásica de trabalho no PLN de Dias-da-Silva et al (2007), a qual prevê o equacionamento das tarefas de PLN em três fases: linguística, representacional e implementacional. A primeira fase consiste na compreensão dos fenômenos linguísticos e na produção de conhecimentos; a segunda fase diz respeito à representação formal (explícita e não ambígua) dos conhecimentos obtidos na fase anterior e, por fim a terceira fase consiste na codificação das representações elaboradas anteriormente em termos de linguagem computacional. Diante dessa metodologia, pode-se dizer que as regras de Ceregatto se inserem na fase representacional, requerendo formalização.

A padronização da descrição das regras (ou formalização) proposta aqui se baseia na Sintaxe de Gherkin. Especificamente, a Sintaxe de Gherkin é uma metodologia que auxilia no desenvolvimento de *software* por equipes colaborativas compostas por membros que não são desenvolvedores (EHRENFRIED, 2019). Essa metodologia consiste em descrever as etapas que o sistema deve seguir utilizando uma série de palavra chaves:

> As palavras-chave mais comuns são:
> • **Dado:** descreve o estado inicial do cenário de teste;
> • **Quando:** descreve a ação que está sendo testada;
> • **E:** adiciona uma nova etapa ao cenário de teste, seguindo a palavra-chave anterior;
> • **Mas:** adiciona uma nova etapa ao cenário de teste, com uma condição ao de exceção
> • **Então:** é a verificação do comportamento do sistema em resposta à ação realizada no passo anterior (CASTRO et.al 2023)

Dessa forma, com base na Sintaxe de Gherkin, fez-se a descrição formal das regras. A título de exemplo, a Regra 2 passou a ser descrita da seguinte forma:

Dado que o *token* corrente é NOUN
Quando este *token* for precedido por outros dois *tokens*
E o primeiro token precedente for NOUN
E o segundo token precedente for DET
Então substituir a etiqueta do *token* corrente NOUN para ADJ

No Quadro 6, apresentam-se as 15 regras de ADJ descritas segundo a Sintaxe de Gherkin.

Quadro 6: Regras de pós-edição para os erros referentes à *tag* ADJ na Sintaxe de Gherkin.

Regra	Descrição	Qt. de casos contemplados	Tipo de regra
1	**Dado** que o token corrente é **ADV** **Quando** este token for a palavra "**certo**" **Então** substituir a etiqueta do token corrente **ADV** para **ADJ**	1	Lexicalizada
2	**Dado** que o token corrente é **NOUN** **Quando** este token for precedido por dois tokens **E** esses tokens são respectivamente **DET** e **NOUN** **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	2	Não-lexicalizada
3	**Dado** que o token corrente é **NOUN** **Quando** este token for precedido por um token e seguido por outro token **E** o token precedente for **DET** **E** o token seguinte for **NOUN** **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Não-lexicalizada
4	**Dado** que o token corrente é **NOUN** **Quando** este token for precedido por um token e seguido por outro token **E** o token precedente for **ADP** **E** o token seguinte for **NOUN** **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Não-lexicalizada
5	**Dado** que o token corrente é **NOUN** **Quando** este token for a palavra "**log**" **E** ser precedido por um token **NOUN** **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Lexicalizada
6	**Dado** que o token corrente é **NOUN** **Quando** este token for a palavra "**casado**" **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Lexicalizada
7	**Dado** que o token corrente é **NOUN** **Quando** este token for a palavra "**coitada**" **E** este token ocupar a primeira posição da sentença **E** este token for seguido de um token **ADP** **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Lexicalizada
8	**Dado** que o token corrente é **NOUN** **Quando** este token for precedido por dois tokens **E** esses tokens são respectivamente **PUNCT** e **ADJ** **E** o token PUNCT for "**/**" **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Não-lexicalizada
9	**Dado** que o token corrente é **NOUN** **Quando** este token for a palavra "**melho**" **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Lexicalizada
10	**Dado** que o token corrente é **NOUN** **Quando** este token for a palavra "**voláteis**" **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Lexicalizada

11	**Dado** que o token corrente é **VERB** **Quando** este token for precedido por dois tokens **E** esses token são respectivamente **ADJ** e **NOUN** **Então** substituir a etiqueta do token corrente **VERB** para **ADJ**	2	Não-lexicalizada
12	**Dado** que o token corrente é **VERB** **Quando** este token for precedido um token **E** esse token for a palavra "**trade**" **Então** substituir a etiqueta do token corrente **VERB** para **ADJ**	1	Lexicalizada
13	**Dado** que o token corrente é **VERB** **Quando** este token for a palavra "**lindo**" **Então** substituir a etiqueta do token corrente **VERB** para **ADJ**	1	Lexicalizada
14	**Dado** que o token corrente é **VERB** **Quando** este token for precedido um token **E** esse token for a palavra "**risco**" **Então** substituir a etiqueta do token corrente **VERB** para **ADJ**	1	Lexicalizada
15	**Dado** que o token corrente é **VERB** **Quando** este token for precedido por três tokens **E** esses tokens são respectivamente **ADP**, **DET** e **NOUN** ou **PROPN** **Então** substituir a etiqueta do token corrente **VERB** para **ADJ**	1	Não-lexicalizada

Fonte: Elaborado pelo autor

O segundo refinamento aqui proposto diz respeito a alterações nas próprias regras, as quais foram submetidas a certos processos, que são: (i) aglutinação, (ii) exceção, (iii) generalização, (iv) exclusão e (v) lexicalização.

Por "aglutinação", entende-se a unificação de regras que têm o mesmo propósito ou lógica semelhante. Esse foi o processo aplicado às Regras 5, 6, 9 e 10, pois, sendo lexicalizadas, buscam corrigir erros de *tagging* de palavras ou *tokens* específicos (como "log", "casado", "melho" e "voláteis"). No caso, todas elas alteram a etiqueta errônea NOUN para ADJ.

Por "exceção", compreende-se a alteração em uma regra que objetiva evitar que ela seja aplicada diante de certos contextos. A Regra 5, por exemplo, prevê que a etiqueta NOUN do *token* "log" seja alterada para ADJ quando a *tag* do *token* anterior for NOUN. Tal condição foi definida porque, no DANTEStocks, "log" pode ser a redução do adjetivo "logarítmico", ocorrendo na sequência de um NOUN. No entanto, percebeu-se a *tag* ADJ não é a única apropriada, pois "log" pode se referir ao nome da empresa "LLX Logística SA", que, no *corpus*, ocorre como LLX LOG (sendo, portanto, PROPN+PROPN). Assim, com o objetivo de tornar a regra mais simples, criou-se uma exceção, sendo preciso apenas que o *token* anterior não seja LLX. Para ilustrar os processos de aglutinação e exceção, tem-se a regra descrita abaixo:

Dado que o *token* corrente é NOUN

Quando este *token* for a palavra "log", "casado", "melho" ou "voláteis"

E ser precedido por um *token*

E esse *token* não é "LLX"

Então substituir a etiqueta do *token* corrente NOUN para ADJ

No que diz respeito à categoria “generalização”, destaca-se que esse processo é o de transformar uma regra lexicalizada para generalizada, utilizando etiquetas PoS ou informações morfológicas. Como exemplo, tem-se as regras 11, 12, 14 e 15, que, antes de serem generalizadas, foram também aglutinadas. A aglutinação, em particular, foi feita ao se observar que os exemplos “risco elevado”, “trade fechado”, “lucro líquido consolidado” e “o bebê abandonado” englobavam adjetivos formados pela forma nominal do verbo chamada particípio passado. Quanto à generalização, ressalta-se que, tendo em vista que as regras 11 e 15 poderiam gerar falsos positivos por utilizarem apenas etiquetas PoS e que as regras 12 e 14 são muito pontuais (lexicalizadas), optou-se por empregar as terminações ou morfemas *-ado* ou *-ido* dos particípios ADJ na generalização. Assim, a regra generalizada que unifica 11, 12, 14 e 15 é:

> **Dado** que o token corrente é VERB
> **Quando** este token tiver os últimos 3 caracteres com a sequência -ado, -ados, -ada, -adas, -ido, -idos, -ida ou -idas
> **E** esse token for diretamente precedido por um token NOUN
> **Então** substituir a etiqueta do token corrente VERB para ADJ

Com relação à categoria “exclusão”, destaca-se que foram excluídas as regras que geravam conflitos com outras, como é o caso, por exemplo, da Regra 3 que compõe o conjunto das 15 cuja análise descrita neste relatório exemplifica o processo de refinamento de todas as regras de Ceregatto (2022). Especificamente, a Regra 3 foi excluída pelo conflito com a Regra 2, ao passo que lidavam com o mesmo padrão linguístico.

E, por fim, o procedimento de “lexicalização” é o inverso da “generalização”, uma vez que as regras originalmente não-lexicalizadas passaram a ser. Tais regras referem-se a erros de etiquetação de palavras de classes fechadas, como a preposição. As 15 regras relativas à *tag* ADJ, utilizadas para exemplificar o processo de análise crítica e validação, não foram alvo do processo de “lexicalização”. Assim, exemplifica-se esse tipo de refinamento com as Regras 20, 21, 22 e 23 que compõem o Apêndice 1 deste documento, as quais buscam corrigir a etiquetação equivocada de preposições (ADP) como SCONJ. Inicialmente, as regras mencionadas não eram lexicalizadas, tendo como condições de aplicação diferentes sequências de etiquetas contextuais à esquerda ou direita do *token*. No entanto, por ADP ser uma classe com 16 palavras (a, ante, até, após, com, contra, de, desde, em, entre, para, perante, por, sem, sob, sobre), torna-se pertinente criar uma regra lexicalizada que altere a etiqueta dessas palavras para ADP sempre que forem classificadas como SCONJ. Assim como:

Dado que o token corrente é SCONJ

Quando este token for “a”, “ante”, “até”, “após”, “com”, “contra”, “de”, “desde”, “em”, “entre”, “para”, “perante”, “por”, “sem”, “sob” ou “sobre”

Então substituir a etiqueta do token corrente SCONJ para ADP.

Embora haja a possibilidade dessa regra gerar falsos positivos, ressalta-se que ela é mais otimizada, pois aglutina 4 regras e soluciona diversos casos de desvios do modelo UDPipe 2.1.

No Quadro 7, ilustram-se os processos de refinamento propostos para as 15 regras referentes à classe ADJ analisada anteriormente.

Quadro 7: Categorização dos refinamentos das regras para pós-edição da *tag* ADJ.

Regra	**Tipo de alteração**
1	Sem alteração
2	Sem alteração
3	Exclusão
4	Sem alteração
5	Aglutinação e Exceção
6	Aglutinação
7	Sem alteração
8	Sem alteração
9	Aglutinação
10	Aglutinação
11	Aglutinação e Generalização
12	Aglutinação e Generalização
13	Sem alteração
14	Aglutinação e Generalização
15	Aglutinação e Generalização

Fonte: Elaborada pelo autor

No Quadro 8, apresenta-se a versão definitiva das regras das 15 regras de ADJ. Devido aos processos de refinamento descritos anteriormente, as regras foram renumeradas. Assim, a antiga Regra 4, por exemplo, passou a ser a Regra 3, uma vez que a antiga Regra 3 foi excluída. Além disso, a nova Regra 4 combina o conteúdo das antigas Regras 5, 6, 7 e 8. Assim, pode-se constatar que houve uma redução na quantidade de regras, de 15 para 8, e a quantidade de casos contemplados também apresentou uma queda de 19 para 18, devido à exclusão da antiga Regra

3. Sendo assim, é possível afirmar que as regras refinadas foram otimizadas em 46%. No entanto, nesse contexto, a otimização não implica em um ganho direto de eficácia na resolução dos problemas de etiquetação automática, mas sim em uma simplificação das regras originais.

Quadro 8 - Regras de pós-edição para os erros referentes à *tag* ADJ após refinamento

Regra	Descrição	Qt. de casos contemplados	Tipo de regra
1	**Dado** que o token corrente é **ADV** **Quando** este token for a palavra "**certo**" **Então** substituir a etiqueta do token corrente **ADV** para **ADJ**	1	Lexicalizada
2	**Dado** que o token corrente é **NOUN** **Quando** este token for precedido por dois tokens **E** esses tokens são respectivamente **DET** e **NOUN** **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	2	Não-lexicalizada
3	**Dado** que o token corrente é **NOUN** **Quando** este token for precedido por um token e seguido por outro token **E** o token precedente for **ADP** **E** o token seguinte for **NOUN** **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Não-lexicalizada
4	**Dado** que o token corrente é **NOUN** **Quando** este token for a palavra "**log**", "**casado**", "**melho**" ou "**voláteis**" **E** ser precedido por um token **E** esse token não é "LLX" **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	4	Lexicalizada
5	**Dado** que o token corrente é **NOUN** **Quando** este token for a palavra "**coitada**" **E** este token ocupar a primeira posição da sentença **E** este token for seguido de um token **ADP** **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Lexicalizada
6	**Dado** que o token corrente é **NOUN** **Quando** este token for precedido por dois tokens **E** esses tokens são respectivamente **PUNCT** e **ADJ** **E** o token PUNCT for "**/**" **Então** substituir a etiqueta do token corrente **NOUN** para **ADJ**	1	Não lexicalizada
7	**Dado** que o token corrente é **VERB** **Quando** este token tiver os últimos 3 caracteres com a sequência -ado, ou -ados, ou -ido, ou -idos **E** esses token for diretamente precedido de **NOUN** **Então** substituir a etiqueta do token corrente **VERB** para **ADJ**	7	Não-lexicalizada
8	**Dado** que o token corrente é **VERB** **Quando** este token for a palavra "**lindo**" **Então** substituir a etiqueta do token corrente **VERB** para **ADJ**	1	Lexicalizada

Fonte: Elaborada pelo autor

Além da formalização das 15 regras para ADJ, o Apêndice 2 engloba a formalização, segundo a Sintaxe de Gherkin, das demais 78 regras de Ceregatto (2022), totalizando as 93 originalmente definidas. No Apêndice 3, tem-se o conjunto final de 68 regras, resultantes do processo de refinamento aqui proposto. Tais regras também estão representadas segundo a Sintaxe de Gherkin. No Apêndice 4, estão as regras originais de Ceregatto que, com base na proposta de refinamento, foram excluídas.

4 AVALIAÇÃO DAS REGRAS

Nesta seção, apresenta-se o procedimento de avaliação das regras de pós-edição. A avaliação teve como objetivo diagnosticar o desempenho das regras originais propostas por Ceregatto, que não haviam sido avaliadas, e compará-lo com o desempenho das regras refinadas neste trabalho. Essa comparação buscou constatar se os refinamentos propostos contribuíam para a etiquetação. Devido ao escopo desta pesquisa, avaliaram-se apenas as 15 regras para ADJ de Ceregatto e as 8 regras para ADJ resultante do refinamento aqui proposto.

Para a avaliação, utilizou-se a ferramenta UDConcord[8] (MIRANDA, PARDO, 2022b), que é um concordanciador que permite listar as ocorrências de determinada palavra ou expressão por meio de uma busca simples pela própria palavra, lema (ou forma canônica), POS *tag*, *deprel* ou *feature.* Dessa forma, diante de uma regra a ser avaliada, buscou-se no *corpus* DANTEStocks pela sua condição (ou descrição, segundo a Sintaxe de Gherkin) de aplicação e, com base nos casos ou ocorrências retornadas, verificou-se se sua aplicação era ou não pertinente. Ressalta-se que a busca foi feita em uma parcela do *corpus* que difere daquela utilizada para produzir as regras (isto é, os pacotes 0-7). No caso, utilizou-se a parcela referente aos demais 5 pacotes de *tweets* (8-12) que compõem o DANTEStocks.

Especificamente, para avaliar a pertinência ou desempenho das regras, aplicou-se a medida ou métrica chamada precisão. De acordo com Silva (2022), a precisão é utilizada para medir o quanto um modelo realiza predições corretamente. No caso, utilizou-se essa métrica para medir o quanto as regras identificam um erro de etiquetação e o corrigi adequadamente. Para tal, é preciso avaliar todas as ocorrências das regras em duas categorias: verdadeiros positivos (VP) e falsos positivos (FP). Os VPs são as ocorrências em que as regras identificam o erro e o corrige corretamente, enquanto os FPs são as ocorrências corrigidas incorretamente pela regra. Considerando VPs e FPs, a precisão é calculada pela equação descrita em (1):

(1) $$Precisão = \frac{VP}{VP+FP}$$

No Quadro 9, dispõem-se os resultados obtidos pelas regras de Ceregatto para a *tag* ADJ. Nele, percebe-se que as Regras 8, 12 e 13 não geraram nenhuma ocorrência, mostrando que a parcela do *corpus* utilizada para teste ou avaliação não possui o tipo de erro coberto pela regra.

Observa-se também que as regras lexicalizadas, como esperado, geraram poucas ocorrências e apresentaram altas precisões, evidenciando o quão pontuais elas são. Por outro

[8] https://udconcord.icmc.usp.br/

lado, as regras não-lexicalizadas resultaram em uma quantidade maior de ocorrências. No entanto, todas as regras desse tipo apresentaram precisões baixas. Por fim, devido às diferenças entre as quantidades de ocorrências identificadas por cada regra para mensurar a precisão do conjunto referente à etiqueta ADJ, fez-se o cálculo da média ponderada das precisões. Dessa forma, pode-se depreender que o conjunto inteiro de regras referentes à etiqueta ADJ possui uma precisão média ponderada de 3,5%.

Quadro 9 – Resultado de desempenho das regras para ADJ de Ceregatto (2022).

Regra	Quantidade de ocorrências encontradas	Verdadeiros positivos	Falsos Positivos	Precisão (%)
1	2	0	2	0
2	50	1	49	2%
3	50	0	50	0%
4	31	0	31	0%
5	1	1	0	100%
6	1	1	0	100%
7	1	1	0	100%
8	0	0	0	0%
9	1	1	0	100%
10	1	1	0	100%
11	21	0	21	0%
12	0	0	0	0%
13	0	0	0	0%
14	3	0	3	0%
15	62	2	61	3,22%
Total	224	8	217	3,50%

Fonte: Elaborada pelo Autor

Ao observar o desempenho das regras refinadas, o qual é apresentado no Quadro 10, o cenário é um pouco diferente. Com base no Quadro 10, as Regras 6 e 8 também não geraram ocorrências, pois elas equivalem às regras originais 8 e 13, respectivamente. A Regra 7, que unificou as antigas 11, 12, 14 e 15, retornou 83 ocorrências, sendo 8 delas VPs, o que indica uma precisão de 9,6%. Ao comparar com a precisão média ponderada das antigas Regras 11, 12, 14 e 15, que era de 2,3%, a nova Regra 7 tem desempenho superior, ainda que a precisão média ponderada de 9,6% seja baixa.

Quadro 10 – Resultado de desempenho das regras para ADJ de Ceregatto refinadas.

Regra	Quantidade de ocorrências encontradas	Verdadeiros Positivos	Falsos Positivos	Precisão (%)
1	2	0	2	0%
2	50	1	49	2%
3	31	0	31	0%
4	4	4	0	100%
5	1	1	0	100%
6	0	0	0	0%
7	83	8	75	9,60%
8	0	0	0	0%
Total	171	14	157	8,1%

Fonte: Elaborada pelo Autor

Além disso, com base no Quadro 10, verifica-se que a nova Regra 4, que resultou da unificação das antigas 6, 7, 9 e 10, não apresentou desempenho diferente das regras originais.

Por fim, a precisão média ponderada do conjunto refinado de regras para ADJ foi de 8,1%, sendo 4,6 pontos percentuais superior às regras originais de Ceregatto (2022). Assim, pode-se dizer que as propostas de refinamento foram positivas, na medida em que é um conjunto 46% menor e 43% mais preciso. Vale salientar, no entanto, que ambos os conjuntos não foram efetivamente implementados no *parser* UDpipe 2.1, o que impede afirmar se a inclusão deles como estratégia de pós-edição melhoria o desempenho do *parser* efetivamente.

5 CONSIDERAÇÕES FINAIS

Neste trabalho, realizou-se uma análise das regras de pós-edição, propostas por Ceregatto (2022), para correção de etiquetações realizadas pelo modelo UDPipe 2.1. Tais regras, no formato no formato lógico (*se, então*), têm como objetivo enriquecer os métodos de *tagging* (estatísticos e/ou probabilísticos, como o UDPipe 2.1) com conhecimento linguístico para textos do tipo CGU (especificamente, *tweets*) em língua portuguesa. Especificamente, as regras de Ceregatto foram propostas a partir da etiquetação automática da parcela de *tweets* do *corpus* DANTEStocks até então revisada manualmente.

Este trabalho, em particular, consistiu em analisar essas regras com o objetivo de propor refinamentos e, na sequência, avaliá-las. A análise permitiu verificar, por exemplo, que a maior parte das regras (61 das 93 iniciais) eram lexicalizadas, enquanto as demais (32 regras) eram

não-lexicalizadas. Com isso, pode-se dizer que o conjunto de regras do autor em questão trata, em sua maioria, de problemas pontuais e de língua geral.

Além disso, diante de alguns problemas nas regras, estas foram submetidas a certos processos, os quais foram classificados em 5 categorias aglutinação, exceção, lexicalização e generalização e exclusão. Por conseguinte, o conjunto de 93 regras foi reduzido para 68 das quais 24 foram aglutinadas, 4 foram lexicalizadas, 2 generalizadas, 3 tiveram exceções e 6 foram excluídas.

Diante de uma limitação de tempo, apenas as regras referentes à *tag* ADJ foram avaliadas. Nessa avaliação, comparou-se o desempenho das 15 regras de Ceregatto com o desempenho das 8 regras resultantes do refinamento proposto neste trabalho. Em suma, quanto a ADJ, a precisão média ponderada das regras originais foi de 3,5%, ao passo que o conjunto refinado de 8 regras obteve 8,1% de precisão. Embora a precisão desses conjuntos seja baixa, verifica-se uma ligeira melhora no desempenho do conjunto refinado.

Diante disso, ressalta-se a necessidade de avaliar o restante das regras refinadas, uma vez que apenas as que se referem à *tag* ADJ foram submetidas a esse processo. Além disso, como trabalho futuro, seria altamente relevante a implementação efetiva de todas as regras refinadas no UDPipe 2.1 como estratégia de pós-edição de *tagging*, a fim de avaliar a eficácia das regras, sobretudo em um *corpus* de CGU diferente do utilizado neste trabalho. A aplicação das regras refinadas em um novo contexto permitiria uma avaliação mais abrangente de sua utilidade e adaptabilidade, além de fornecer conhecimentos valiosos para aprimorar ainda mais seu desempenho.

Por fim, o desenvolvimento deste Trabalho de Conclusão de Curso possibilitou ao aluno explorar o campo da etiquetação automática de *part-of-spech* para textos de CGU. Mesmo sendo a tarefa de *tagging* um tópico visto em algumas disciplinas de PLN na graduação, salienta-se que sua aplicação a textos do tipo CGU, no entanto, é recente e, por meio deste TCC, foi possível se familiarizar com esse cenário. Assim, pode-se dizer que o desenvolvimento do TCC permitiu aprofundar os conhecimentos em PLN, adquirindo uma compreensão mais sólida das técnicas e algoritmos envolvidos na etiquetação de PoS. Além disso, a análise das regras no referido contexto permitiu exercitar a habilidade crítica sobre a tarefa de descrição linguística e de avaliação no PLN. Com isso, este trabalho enriqueceu a formação do graduando.

6 REFERÊNCIAS BIBLIOGRÁFICAS

BEHZAD, S.; ZELDES, A. A cross-genre ensemble approach to robust Reddit part of speech tagging. In: WEB AS CORPUS WORKSHOP, 12, 2020, Marseille. **Proceedings**… Marseille: ACL, 2020. p. 50-56.

BOHNET, B. et al. Morphosyntactic tagging with a meta-BiLSTM model over context sensitive token encodings. In: ANNUAL MEETING OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS, 56, 2018, Melbourne. **Proceedings** [...]. Melbourne: ACL, 2018. p. 2642–2652.

BOSCO, C. et al. Overview of the EVALITA 2016 Part Of Speech Tagging on TWitter for ITAlian task. In: EVALUATION CAMPAIGN OF NATURAL LANGUAGE PROCESSING AND SPEECH TOOLS FOR ITALIAN, 5, 2016, Naples. **Proceedings** […]. Naples: CEUR, 2016, p. 1-7.

CASTRO, A. P. F. B. HERCULINO, G. C. S. MENDONÇA, V. H. B. SILVA, C. **A. Padronização e reciclagem de códigos em um Marketplace utilizando Gherkin: um estudo de caso na AgroBusiness com testes automatizados RSpec, Capybara e Page Objects**. Curso de Bacharelado em Sistemas de Informac ̧ao – Faculdade de Computação (FACOM). Universidade Federal de Mato Grosso do Sul (UFMS).

CEREGATTO, Gabriel**. Caracterização Morfossintática de um Corpus de Tweets e Análise Preliminar de Erros de Tagging**. 2022 - São Carlos. Trabalho de Conclusão de Curso (TCC) - Universidade Federal de São Carlos - UFSCar.

DEVLIN, J. et al. BERT: Pre-training of deep bidirectional transformers for language understanding. In: CONFERENCE OF THE NORTH AMERICAN CHAPTER OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS: HUMAN LANGUAGE TECHNOLOGIES, 2016, Minneapolis. **Proceedings** […]. Minneapolis, 2019, p. 4171-4186.

DI-FELIPPO, A. et al. Descrição preliminar do corpus DANTEStocks: diretrizes de segmentação para anotação segundo Universal Dependencies. In: JORNADA DE DESCRIÇÃO DO PORTUGUÊS, 7, 2021. **Anais** […]. 2021, p. 335-343. (evento online)

DI-FELIPPO, A. et al. Diretrizes de anotação de tag PoSs em tweets do mercado financeiro: orientações para anotação em Língua Portuguesa segundo a abordagem Universal Dependencies**. Relatório Técnico do ICMC**, 438. ICMC, USP. São Carlos-SP, 24p, 2022.

DOMINGUES, M. L. C. S. **Abordagem para o desenvolvimento de um etiquetador de alta acurácia para o português do Brasil**. 2011. Tese (Pós-graduação) - Universidade Federal do Pará (UFPA). Acesso em: 22 de agosto de 2023.

DURAN, M.S. Manual de anotação de tag PoSs: orientações para anotação de etiquetas morfossintáticas em Língua Portuguesa, seguindo as diretrizes da abordagem Universal Dependencies. **Relatório Técnico do ICMC**, 434. ICMC, USP, São Carlos, 55p, 2021

EHRENFRIED, Henrique Varella. **Gherkin Specification Extension - Uma Linguagem de Especificação de Requisitos Baseada em Gherkin**. 2019. Tese (Mestrado) - Universidade Federal do Paraná - Curitiba. Acesso em: 22 de agosto de 2023.

GUIBON, G., et al. When collaborative treebank curation neets graph grammars: Arborator with a grew back-end. In: INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION CONFERENCE, 12, 2012, Istanbul. **Proceedings** […]. Istanbul: ELRA, 2012, p. 5291-5300.

JURAFSKY, D.; MARTIN, J.H. **Speech and Language Processing: an introduction to Natural Language Processing, Computational Linguistics and Speech Recognition**. 3rd ed. Available at: https://web.stanford.edu/~jurafsky/slp3/. Access in: 1 July 2022.

KEPLER, F. N. **Modelagem de contextos para aprendizado automático aplicado à Análise Morfossintática**. Tese (Doutorado) — Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo, SP, Brasil, 2010.

KLEIN, S.; SIMMONS, R. F. A computational approach to grammatical coding of english words. **Association for Computing Machinery**, p. 334–347, 1963.
KONDRATYUK, D.; STRAKA, M. 75 languages, 1 model: parsing Universal Dependencies universally. In: CONFERENCE ON EMPIRICAL METHODS IN NATURAL LANGUAGE PROCESSING, 2019, Hong Kong. **Proceedings** […]. Hong Kong: ACL, 2019. p. 2779–2795.

LYNN, T. et al. Minority language Twitter: part-of-speech tagging and analysis of Irish tweets. In: WORKSHOP ON NOISY USER-GENERATED TEXT, 2015, Beijing. **Proceedings** […]. Beijing: ACL, 2015, p. 1-8.

MIRANDA, L.G.M., PARDO, T.A.S. An improved and extended annotation tool for Universal Dependencies-based treebank construction. In: INTERNATIONAL CONFERENCE ON THE COMPUTATIONAL PROCESSING OF PORTUGUESE - DEMO WORKSHOP, 2022. **Proceedings** […]. 2022a, p. 1-3. (Evento online)

MIRANDA, L.G.M.; PARDO, T.A.S. (2022). UDConcord: A Concordancer for Universal Dependencies Treebanks. In: UNIVERSAL DEPENDENCIES BRAZILIAN FESTIVAL (UDFest-BR), 1, 2022b. Proceedings […]. 2022. pp. 1-10. (online event)

MITKOV, R. (Ed.) **The Oxford Handbook of Computational Linguistics** (Oxford Handbooks in Linguistics S.), 1st ed. USA: Oxford University Press, Inc., 2005.

NIVRE, J. Towards a Universal Grammar for Natural Language Processing. In: Gelbukh, A. (Eds) Computational Linguistics and Intelligent Text Processing. CICLing 2015. **Lecture Notes in Computer Science**, vol 9041. Springer, p. 3-16. https://doi.org/10.1007/978-3-319-18111-0_1.

NIVRE, J. et al. Universal Dependencies v1: a multilingual treebank collection. In: INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION,10, 2016, Portorož. **Proceedings** […]. Portorož: ELRA, 2016. p.1659-66.

NIVRE, J., MARIE-CATHERINE. M., GINTER, F. et al. Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection. In: INTERNATIONAL CONFERENCE ON

LANGUAGE RESOURCES AND EVALUATION, 12, 2020, Marseille. **Proceedings** […]. Marseille: ELRA, 2020, p. 4034-4043.

PROISL, T. Someweta: a part-of-speech tagger for German social media and web texts. In: INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION, 11, 2018, Miyazaki. **Proceedings** […]. Miyazaki: ELRA, 2018, p. 665-670.
REHBEIN, I., RUPPENHOFER, J., BICH-NGOC, D. tweeDe – a Universal Dependencies treebank for German tweets. In: INTERNATIONAL WORKSHOP ON TREEBANKS AND LINGUISTIC THEORIES, 18, 2019, Paris. **Proceedings** […]. Paris: ACL, 2019, p. 100-108.

SANGUINETTI, M. et al. PoSTWITA-UD: an Italian Twitter Treebank in Universal Dependencies. In: INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION, 11, 2018. Miyazaki. **Proceedings** […]. Miyazaki: ELRA, 2018, p. 1768-1775

SILVA, F. J. V.; ROMAN, N. T.; CARVALHO, A.M.B.R. Stock market tweets annotated with emotions. **Corpora**, 15(3), p. 343-54, 2020. Online ISSN 1755-1676.

SILVA, E.H**. Etiquetagem morfossintática multigênero para o português do Brasil segundo o modelo Universal Dependencies**. Qualificação (Mestrado em Ciências de Computação e Matemática Computacional) - Universidade de São Paulo, São Carlos, SP, Brasil, 2022.

STRAKA, M.; HAJIČ, J.; STRAKOVÁ, J. UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In: INTERNATIONAL CONFERENCE ON LANGUAGE RESOURCES AND EVALUATION, 10, 2016, Portorož, **Proceedings** [...]. Portorož: ELRA, 2016. p. 4290-4297.

STRAKOVÁ, J.; HAJIC, J. UDPipe at SIGMORPHON 2019: contextualized embeddings, regularization with morphological categories, corpora merging. In: WORKSHOP ON COMPUTATIONAL RESEARCH IN PHONETICS, PHONOLOGY, AND MORPHOLOGY, 16, 2019, Florence. **Proceedings** [...]. Florence, 2019. p. 95– 103

APÊNDICE 1 – REGRAS DE PÓS-EDIÇÃO E CARACTERIZAÇÃO DOS ERROS DE *TAGGING* (CEREGATTO, 2022)

Qt	DE	PARA	Condição	Regra	Exemplo	Qt	Tipo de regra	Linguagem	Fenômeno CGU
1	ADV	ADJ	Se token="certo"	Substituir ADV por ADJ	certo/ADJ>ADV	1	Lexicalizada	Língua geral	
2	NOUN	ADJ	Se o token for precedido por DET e NOUN	Substituir por NOUN por ADJ	em/ADP a/DET sexta/NOUN **passada**/ADJ>NOUN	2	Não-lexicalizada	Língua geral	
3	NOUN	ADJ	Se o token for imediatamente precedido por DET e sucedido por NOUN.	Substituir NOUN por ADJ	a/DET **corrente**/NOUN>ADJ gravidade/NOUN	1	Não-lexicalizada	Língua geral	
4	NOUN	ADJ	Se o token for imediatamente precedido por ADP e sucedido por NOUN.	Substituir NOUN por ADJ	de/ADP **módico**/NOUN>ADJ repique/NOUN	1	Não-lexicalizada	Língua geral	
5	NOUN	ADJ	Se o token anterior for NOUN, substituir tag do token "log" por ADJ.	Substituir NOUN por ADJ	escala/NOUN **log**/ADJ>NOUN	1	Lexicalizada	Domínio	
6	NOUN	ADJ	Se token="casado"	Substituir NOUN por ADJ	**casado/ADJ>NOUN**	1	Lexicalizada	Língua geral	
7	NOUN	ADJ	Se token="coitada", primeiro token da sentença e seguido por ADP	Substituir NOUN por ADJ	**coitada/ADJ>NOUN**	1	Lexicalizada	Língua geral	
8	NOUN	ADJ	Se o token for seguido por PUNCT (/) + ADJ	Substituir NOUN por ADJ	**médio/ADJ>NOUN />PUNCT longo/NOUN**	1	Não-lexicalizada	Língua geral	
9	NOUN	ADJ	Se token="melho"	Substituir qualquer tag PoS por ADJ	**melho/NOUN>ADJ**	1	Lexicalizada	Língua geral	Truncamento
10	NOUN	ADJ	Se token="voláteis"	Substituir qualquer tag PoS por ADJ	**voláteis/NOUN>ADJ**	1	Lexicalizada	Língua geral	
11	VERB	ADJ	Se token 1E e 2E = ADJ + NOUN	Substituir VERB por ADJ	lucro/NOUN líquido/ADJ **consolidado**/ADJ>VERB	2	Não-lexicalizada	Língua geral	
12	VERB	ADJ	Se token 1E="trade"	Substituir VERB por ADJ	Trade/NOUN **fechado**/ADJ>VERB	1	Lexicalizada	Língua geral	
13	VERB	ADJ	Se token="lindo"	Substituir qualquer tag PoS por ADJ	lindo/ADJ>VERB	1	Lexicalizada	Língua geral	
14	VERB	ADJ	Se token 1E="risco"	Substituir VERB por ADJ	risco/NOUN **elevado**/ADJ>VERB	1	Lexicalizada	Língua geral	
15	VERB	ADJ	Se tokens 1D, 2D e 3D = ADP + DET + NOUN/PROPN	Substituir VERB por ADJ	para/ADP o/DET bebê/NOUN **abandonado**/ADJ>VERB	3	Não-lexicalizada	Língua geral	
16	ADV	ADP	Se token 1D="assim", trocar ADV por ADP	Substituir ADV por ADP	assim/ADV **como**/ADP fizemos/VERB	1	Lexicalizada	Língua geral	
17	DET	ADP	Se token 1E=VERB, substituir DET por ADP	Substituir DET por ADP	que/PRON correspondem/VERB **a**/ADP	1	Lexicalizada	Língua geral	Truncamento
18	NOUN	ADP	Se token 1D="market", substituir qualquer tag que não seja ADP por ADP	Substituir qualquer tag PoS por ADP	o/DET **after**/ADP market/NOUN	1	Lexicalizada	Domínio	Estrang.
19	NOUN	ADP	Se token = "d", substituir qualquer tag que não seja ADP por ADP	Substituir qualquer tag PoS por ADP	42/NUM **d**/ADP	1	Lexicalizada	Língua geral	Truncamento
20	SCONJ	ADP	Se token 1D=SCONJ, substituir SCONJ por ADP	Substituir SCONJ por ADP	perspectiva/NOUN **de**/ADP que/SCONJ	2	Não-lexicalizada	Língua geral	

21	SCONJ	ADP	Se 1E=NOUN e 1D=VERB, substituir SCONJ por ADP	Substituir SCONJ por ADP	Ações/NOUN para/ADP comprar/VERB	2	Não-lexicalizada	Língua geral	
22	SCONJ	ADP	Se 1E=NOUN e 2D for VERB, substituir SCONJ por ADP	Substituir SCONJ por ADP	hora/NOUN **de**/ADP ela/PRON desabar/VERB	3	Não-lexicalizada	Língua geral	
23	SCONJ	ADP	Se NOUN + ADJ em 2E e 1E, e VERB em 1D, substituir SCONJ por ADP	Substituir SCONJ por ADP	fundo/NOUN soberano/ADJ para/ADP comprar/VERB	1	Não-lexicalizada	Língua geral	
24	ADP	ADV	Se token 1D='vc'	Substituir ADP por ADV	**como/ADV** vc/PRON	1	Lexicalizada	Domínio	
25	ADP	ADV	Se token 1D='mesmo'	Substituir ADP por ADV	**até/ADV mesmo/ADV**	1	Lexicalizada	Língua geral	
26	NOUN	ADV	Se token= "tb"	Substituir NOUN por ADV	**tb/ADV**	1	Lexicalizada	Domínio	Abreviação
27	NOUN	ADV	Se token='menos'	Substituir NOUN por ADV	**menos/ADV**	1	Lexicalizada	Língua geral	
28	NOUN	ADV	Se token='avante'	Substituir NOUN por ADV	**avante/ADV**	1	Lexicalizada	Língua geral	
29	NOUN	ADV	Se token='cm'	Substituir NOUN por ADV	**cm/ADV**	1	Lexicalizada	Domínio	Abreviação
30	NOUN	ADV	Se token='msm'	Substituir NOUN por ADV	**msm/ADV**	1	Lexicalizada	Domínio	Abreviação
31	NOUN	ADV	Se token 1E=ADP	Substituir NOUN por ADV	para/ADP **baixo/ADV**	2	Lexicalizada	Língua geral	
32	PRON	ADV	Se token 1D='nojo'	Substituir PRON por ADV	**QUE/ADV nojo/NOUN**	1	Lexicalizada	Língua geral	
33	PRON	ADV	Se token='pouco' antecedido de VERB	Substituir PRON por ADV	há/VERB **pouco/ADV**	1	Lexicalizada	Língua geral	
34	SCONJ	ADV	Se primeiro token='como' for primeiro token da sentença seguido por VERB	Substituir SCONJ por ADV	**Como/ADV diria/VERB**	1	Lexicalizada	Língua geral	
35	SCONJ	ADV	Se 1D for token='que'	Substituir SCONJ por ADV	**tanto/ADV que/SCONJ**	2	Lexicalizada	Língua geral	
36	VERB	ADV	Se token='fora'	Substituir VERB por ADV	**fora/ADV**	2	Lexicalizada	Língua geral	
37	ADJ	DET	se existir DET em 1E	Substituir ADJ por DET	o/DET **mesmo/DET**	2	Não-lexicalizada	Língua geral	
38	ADP	DET	Se 1E e 1D forem NOUN	Substituir ADP por DET	fechamento/NOUN **de/DET** gap/NOUN	2	Não-lexicalizada	Língua geral	
39	ADP	DET	Se 1E=VERB e 1D=NOUN	Substituir ADP por DET	comece/VERB **a/DET** volatil/NOUN	2	Não-lexicalizada	Língua geral	
40	PRON	DET	Se PUNCT em 1E ou 2E	Substituir PRON por DET	./PUNCT **Esse/DET**	2	Não-lexicalizada	Língua geral	
41	PRON	DET	Se DET em 1E ou 2E	Substituir PRON por DET	o/DET **tal/DET**	2	Não-lexicalizada	Língua geral	
42	SCONJ	DET	Se token='que' seguido de token= 'ganhos'	Substituir SCONJ por DET	**que/DET ganhos/NOUN**	1	Lexicalizada	Língua geral	
43	SCONJ	DET	Se token= 'QUE' seguido de token='TAL'	Substituir SCONJ por DET	**QUE/DET TAL/PRON**	1	Lexicalizada	Língua geral	
44	ADJ	INTJ	Se existir PUNCT em 1D	Substituir ADJ por INTJ	**Pronto/INTJ** !/PUNCT	2	Não-lexicalizada	Língua geral	
45	ADV	INTJ	Se token='né'	Substituir ADV por INTJ	**né/INTJ**	1	Lexicalizada	Domínio	Coloquialismo
46	ADV	INTJ	Se token='amém'	Substituir ADV por INTJ	**amém/INTJ**	1	Lexicalizada	Língua geral	
47	ADV	INTJ	Se token='Ah'	Substituir ADV por INTJ	**Ah/INTJ**	1	Lexicalizada	Língua geral	
48	AUX	INTJ	Se existir PUNCT em 1D	Substituir ADJ por INTJ	**É/INTJ ,/PUNCT**	1	Não-lexicalizada	Língua geral	
49	NOUN	INTJ	Se token='blz'	Substituir NOUN por INTJ	**blz/INTJ**	1	Lexicalizada	Domínio	Abreviação
50	NOUN	INTJ	Se token='Ops'	Substituir NOUN por INTJ	**Ops/INTJ**	1	Lexicalizada	Domínio	Coloquialismo
51	PRON	INTJ	Se token='ô'	Substituir AUX por INTJ	**ô/INTJ**	1	Lexicalizada	Língua geral	Coloquialismo

52	ADJ	NOUN	Se token="gasolina".	Então substituir qualquer PoS que não seja NOUN por NOUN.	bolsa/NOUN **gasolina**/ADJ>NOUN	1	Lexicalizada	Língua geral	
53	ADJ	NOUN	Se token="B" e pos do 1º token E for "plano".	Então substituir pos diferente de NOUN por NOUN.	plano/NOUN **B**/ADJ>NOUN	1	Lexicalizada	Língua geral	
54	ADJ	NOUN	Se pos=ADJ.	Então substituir ADJ por NOUN se pos do 1º token E for DET e do 1º token D não for NOUN	a/DET **segunda**/ADJ>NOUN se/PRON	1	Não-lexicalizada	Língua geral	
55	ADJ	NOUN	Se pos=ADJ e pos do 1º token E for VERB e do 1º token à direira for ADP.	Então substituir ADJ por NOUN.	renova/VERB **mínima**/ADJ>NOUN de/ADP	1	Não-lexicalizada	Língua geral	
56	ADJ	NOUN	Se um dos ordinais 2ª, 3ª, 4ª, 5ª ou 6ª=ADJ e pos do 2º token E for ADP e o 1º token E for "esta/DET".	Então substituir ADJ por NOUN.	em/ADP esta/DET **5ª**/ADJ>NOUN	1	Lexicalizada	Língua geral	
57	ADJ	NOUN	Se os tokens "acionista"/"acionistas"=ADJ.	Então substituir ADJ para NOUN.	ser/AUX **acionista**/ADJ>NOUN	1	Lexicalizada	Língua geral	
58	ADJ	NOUN	Se token "estatal" ou "estatais"=ADJ.	Então substituir ADJ por NOUN se pos do 1º token E não for NOUN.	foram/AUX só/ADV **estatais**/ADJ>NOUN	1	Lexicalizada	Língua geral	
59	ADJ	NOUN	Se pos=ADJ.	Então substituir ADJ por NOUN se pos do 1º token E for DET e do 1º token D não for NOUN	minha/DET **querida**/ADJ>NOUN !/PUNCT	1	Não-lexicalizada	Língua geral	
60	ADJ	NOUN	Se sequência de pos=ADJ+NOUN e pos do 1º token E de ADJ for DET e do 1º token D de NOUN for DET ou CCONJ e do 2º D de NOUN for ADJ.	Então inverter as tags (NOUN+ADJ).	a/DET **sexta**/ADJ>NOUN **passada**/NOUN>ADJ uma/DET sub/ADJ	1	Não-lexicalizada	Língua geral	
61	ADJ	NOUN	Se sequência de pos=ADJ+NOUN e pos do 1º token E de ADJ for DET e do 1º token D de NOUN for DET ou CCONJ e do 2º D de NOUN for ADJ.	Então inverter as tags (NOUN+ADJ).	a/DET **alta**/ADJ>NOUN **esquisita**/NOUN>ADJ e/CCONJ repentina/ADJ	1	Não-lexicalizada	Língua geral	
62	PRON	NOUN	Se token "el"=PRON e um dos 2 tokens prévios (E) for "perder"/VERB.	Então substituir PRON por NOUN.	perder/VERB a/DET **el**/PRON>NOUN	1	Lexicalizada	Domínio	Truncamento
63	SYM	NOUN	Se token "abs"=SYM.	Então substituir SYM por NOUN	?/PUNCT **Abs**/SYM>NOUN	1	Lexicalizada	Domínio	Abreviação
64	VERB	NOUN	Se token "bela"=PROPN + token "escolha"=VERB.	Então substituir PROPN por ADJ e VERB por NOUN.	**Bela/PROPN>ADJ escolha/VERB>NOUN**	1	Lexicalizada	Língua geral	
65	VERB	NOUN	Se pos=VERB e pos do 1º token E for PROPN e 1º token D for ":/PUNCT".	Então substituir VERB por NOUN.	#petr4/PROPN **Recompra**/VERB>NOUN :/PUNCT	1	Lexicalizada	Língua geral	
66	VERB	NOUN	Se pos=VERB e pos do 1º token E não for PRON.	Então substituir VERB por NOUN.	@JHF_Oficial/PROPN **err**o/VERB>NOUN	1	Não-lexicalizada	Língua geral	
67	VERB	NOUN	Se token "desculpa"=VERB e pos do 1º token E não for PRON.	Então substituir VERB por NOUN.	peço/NOUN sempre/ADV **desculpa**/VERB>NOUN	1	Lexicalizada	Língua geral	
68	VERB	NOUN	Se token 1E="que".	Então substituir VERB por NOUN.	que/DET **ganhos**/VERB>NOUN	1	Lexicalizada	Língua geral	
69	VERB	NOUN	Se pos=VERB e pos do 2º token E for NOUN e do 1º token E for CCONJ e do 1º token D ser ADP.	Então substituir VERB por NOUN.	baixa/NOUN e/CCONJ **agulhada**/VERB>NOUN de/ADP	1	Não-lexicalizada	Língua geral	
70	VERB	NOUN	Se pos=VERB e pos do 1º token D for ADP e do 2º token D for PROPN.	Então substituir VERB por NOUN.	**repique/VERB>NOUN de/ADP oibr4/PROPN**	1	Não-lexicalizada	Língua geral	
71	ADV	PRON	Se token = "algo"	substituir qualquer tag PoS por PRON	**algo/PRON raro/ADJ**	1	Lexicalizada	Língua geral	

72	ADV	PRON	Se token 1E = "QUE"	Substituir ADV por PRON	QUE/DET **TAL/PRON**	1	Lexicalizada	Língua geral	
73	DET	PRON	Se token = "o" e token 1D = "q"	Substituir DET por PRON	**o/PRON q/PRON**	2	Lexicalizada	Língua geral	
74	DET	PRON	Se 1D = NOUN	Substituir DET por PRON	**Esse/PRON tb/ADV**	1	Não-lexicalizada	Língua geral	
75	NOUN	PRON	Se token = "vcs"	substituir qualquer tag PoS que não seja PRON por PRON	**vcs/PRON>NOUN**	2	Lexicalizada	Domínio	Abreviação
76	NOUN	PRON	Se token = "mim	substituir qualquer tag PoS que não seja PRON por PRON	**mim/PRON>NOUN**	1	Lexicalizada	Língua geral	Abreviação
77	ADP	SCONJ	Se ADV/VERB em 1E, 2E ou 3E e VERB/AUX em 1D, 2D ou 3D	Substituir ADP por SCONJ	antes/ADV **de/SCONJ** ser/AUX	1	Não-lexicalizada	Língua geral	
78	ADV	SCONJ	Se VERB em 2E ou 1E	Substituir ADV por SCONJ	ver/VERB **como/SCONJ**	2	Não-lexicalizada	Língua geral	
79	PRON	SCONJ	Se VERB em 1D, 2D ou 3D	Substituir PRON por SCONJ	**q/SCONJ** vcs/PRON conhecem/VERB	3	Não-lexicalizada	Língua geral	
80	NOUN	SYM	Se token= 'x'	Substituir NOUN por SYM	**x/SYM**	1	Lexicalizada	Domínio	Símbolo
81	ADJ	VERB	Se token= 'regist'	Substituir ADJ por VERB	**registr/VERB**	1	Lexicalizada	Domínio	Truncamento
82	ADJ	VERB	Se token= 'fazen'	Substituir ADJ por VERB	**fazen/VERB**	1	Lexicalizada	Domínio	Truncamento
83	ADP	VERB	Se ADV em 1E	Substituir ADP por VERB	não/ADV **para/VERB**	1	Não-lexicalizada	Língua geral	
84	ADP	VERB	Se houver CCONJ em 1D	Substituir ADP por VERB	**alugado/VERB** mas/CCONJ	1	Não-lexicalizada	Língua geral	
85	ADP	VERB	Se token= 'há'	Substituir ADP por VERB	**há/VERB**	1	Lexicalizada	Língua geral	
86	AUX	VERB	Se não houver VERB em 1D, 2D ou 3D	Substituir AUX por VERB	quem/PRON **for/VERB** a/DET faixa/NOUN	4	Não-lexicalizada	Língua geral	
87	NOUN	VERB	Se token= 'monitoro'	Substituir NOUN por VERB	**monitoro/VERB**	1	Lexicalizada	Língua geral	
88	NOUN	VERB	Se token= 'Lembra'	Substituir NOUN por VERB	**Lembra/VERB**	1	Lexicalizada	Língua geral	
89	NOUN	VERB	Se token= 'renova'	Substituir NOUN por VERB	**renova/VERB**	1	Lexicalizada	Língua geral	
90	NOUN	VERB	Se token= 'Trabalho' sem DET em posição 1D	Substituir NOUN por VERB	**@pagina2/PROPN Trabalho/VERB**	1	Lexicalizada	Língua geral	
91	NOUN	VERB	Se token= 'venda' seguido de tokens 'de' e 'novo'	Substituir NOUN por VERB	**venda/VERB de/ADP novo/NOUN**	1	Lexicalizada	Língua geral	
92	NOUN	VERB	Se token ='peço'	Substituir NOUN por VERB	**peço/VERB**	1	Lexicalizada	Língua geral	
93	NOUN	VERB	Se token='partir'	Substituir NOUN por VERB	**partir/VERB**	1	Lexicalizada	Língua geral	

APÊNDICE 2 – FORMALIZAÇÃO NA SINTAXE DE GHERKIN

Regra	DE	PARA	Descrição	Qt.	Tipo de regra	Linguagem	Fenômeno CGU
Regra 1	ADV	ADJ	Dado que o token corrente é ADV Quando este token for a palavra “certo” Então substituir a etiqueta do token corrente ADV para ADJ	1	Lexicalizada	Língua geral	
Regra 2	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for precedido por dois tokens E esses tokens são respectivamente DET e NOUN Então substituir a etiqueta do token corrente NOUN para ADJ	2	Não-lexicalizada	Língua geral	
Regra 3	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for precedido por um token e seguido por outro token E o token precedente for DET E o token seguinte for NOUN Então substituir a etiqueta do token corrente NOUN para ADJ	1	Não-lexicalizada	Língua geral	
Regra 4	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for precedido por um token e seguido por outro token E o token precedente for ADP E o token seguinte for NOUN Então substituir a etiqueta do token corrente NOUN para ADJ	1	Não-lexicalizada	Língua geral	
Regra 5	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for a palavra “log” E ser precedido por um token NOUN Então substituir a etiqueta do token corrente NOUN para ADJ	1	Lexicalizada	Domínio	
Regra 6	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for a palavra “casado” Então substituir a etiqueta do token corrente NOUN para ADJ	1	Lexicalizada	Língua geral	
Regra 7	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for a palavra “coitada” E este token ocupar a primeira posição da sentença E este token for seguido de um token ADP Então substituir a etiqueta do token corrente NOUN para ADJ	1	Lexicalizada	Língua geral	
Regra 8	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for precedido por dois tokens E esses tokens são respectivamente PUNCT e ADJ E o token PUNCT for “/” Então substituir a etiqueta do token corrente NOUN para ADJ	1	Não-lexicalizada	Língua geral	
Regra 9	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for a palavra “melho” Então substituir a etiqueta do token corrente NOUN para ADJ	1	Lexicalizada	Língua geral	Truncamento
Regra 10	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for a palavra “voláteis” Então substituir a etiqueta do token corrente NOUN para ADJ	1	Lexicalizada	Língua geral	

Regra 11	VERB	ADJ	Dado que o token corrente é VERB Quando este token for precedido por dois tokens E esses token são respectivamente ADJ e NOUN Então substituir a etiqueta do token corrente VERB para ADJ	2	Não-lexicalizada	Língua geral	
Regra 12	VERB	ADJ	Dado que o token corrente é VERB Quando este token for precedido um token E esse token for a palavra “trade” Então substituir a etiqueta do token corrente VERB para ADJ	1	Lexicalizada	Língua geral	
Regra 13	VERB	ADJ	Dado que o token corrente é VERB Quando este token for a palavra “lindo” Então substituir a etiqueta do token corrente VERB para ADJ	1	Lexicalizada	Língua geral	
Regra 14	VERB	ADJ	Dado que o token corrente é VERB Quando este token for precedido um token E esse token for a palavra “risco” Então substituir a etiqueta do token corrente VERB para AD	1	Lexicalizada	Língua geral	
Regra 15	VERB	ADJ	Dado que o token corrente é VERB Quando este token for precedido por três tokens E esses tokens são respectivamente ADP, DET e NOUN ou PROPN Então substituir a etiqueta do token corrente VERB para ADJ	3	Não-lexicalizada	Língua geral	
Regra 16	ADV	ADP	Dado que o token corrente é ADV Quando este token for precedido por um token E esse token for “assim” Então substituir a etiqueta do token corrente ADV para ADP	1	Lexicalizada	Língua geral	
Regra 17	DET	ADP	Dado que o token corrente é DET Quando este token for precedido por um token E esse token for VERB Então substituir a etiqueta do token corrente DET para ADP	1	Lexicalizada	Língua geral	Truncamento
Regra 18	NOUN	ADP	Dado que o token corrente não é ADP Quando este token for precedido por um token E esse token for “market” Então substituir a etiqueta do token corrente para ADP	1	Lexicalizada	Domínio	Estrang.
Regra 19	NOUN	ADP	Dado que o token corrente não é ADP Quando este token for “d” Então substituir a etiqueta do token corrente para ADP	1	Lexicalizada	Língua geral	Truncamento
Regra 20	SCONJ	ADP	Dado que o token corrente é SCONJ Quando este token for precedido por um token E esse token for SCONJ Então substituir a etiqueta do token corrente SCONJ para AD	2	Não-lexicalizada	Língua geral	
Regra 21	SCONJ	ADP	Dado que o token corrente é SCONJ Quando este token for precedido por um token e seguido por outro token E o token precedente for NOUN E o token seguinte for VEB Então substituir a etiqueta do token corrente SCONJ para ADP	2	Não-lexicalizada	Língua geral	

Regra 22	SCONJ	ADP	Dado que o token corrente é SCONJ Quando este token for precedido por um token e seguido por outros dois tokens E o token precedente for NOUN E o primeiro token seguinte for de qualquer classe E o segundo token seguinte for VERB Então substituir a etiqueta do token corrente SCONJ para ADP	3	Não-lexicalizada	Língua geral	
Regra 23	SCONJ	ADP	Dado que o token corrente é SCONJ Quando este token for precedido por dois tokens e seguido por outro token E o segundo token precedente for NOUN E o primeiro token precedente for ADJ E o segundo token seguinte for VERB Então substituir a etiqueta do token corrente SCONJ para ADP	1	Não-lexicalizada	Língua geral	
Regra 24	ADP	ADV	Dado que o token corrente é ADP Quando este token for seguido por um token E esse token for “vc” Então substituir a etiqueta do token corrente ADP para ADV	1	Lexicalizada	Língua geral	
Regra 25	ADP	ADV	Dado que o token corrente é ADP Quando este token for seguido por um token E esse token for “mesmo” Então substituir a etiqueta do token corrente ADP para ADV	1	Lexicalizada	Língua geral	
Regra 26	NOUN	ADV	Dado que o token corrente é NOUN Quando este token for “tb” Então substituir a etiqueta do token corrente NOUN para ADV	1	Lexicalizada	Domínio	Abreviação
Regra 27	NOUN	ADV	Dado que o token corrente é NOUN Quando este token for “menos” Então substituir a etiqueta do token corrente NOUN para ADV	1	Lexicalizada	Língua geral	
Regra 28	NOUN	ADV	Dado que o token corrente é NOUN Quando este token for “avante” Então substituir a etiqueta do token corrente NOUN para ADV	1	Lexicalizada	Língua geral	
Regra 29	NOUN	ADV	Dado que o token corrente é NOUN Quando este token for “cm” Então substituir a etiqueta do token corrente NOUN para ADV	1	Lexicalizada	Domínio	Abreviação
Regra 30	NOUN	ADV	Dado que o token corrente é NOUN Quando este token for “msm” Então substituir a etiqueta do token corrente NOUN para ADV	1	Lexicalizada	Domínio	Abreviação
Regra 31	NOUN	ADV	Dado que o token corrente é NOUN Quando este token for precedido por um token E esse token for ADP Então substituir a etiqueta do token corrente NOUN para ADV	2	Lexicalizada	Língua geral	
Regra 32	PRON	ADV	Dado que o token corrente é PRON Quando este token for precedido por um token E esse token for “nojo” Então substituir a etiqueta do token corrente PRON para ADV	1	Lexicalizada	Língua geral	

Regra 33	PRON	ADV	Dado que o token corrente é PRON Quando este token for “pouco” E estiver precedido de um token E esse token for VERB Então substituir a etiqueta do token corrente PRON para ADV	1	Lexicalizada	Língua geral	
Regra 34	SCONJ	ADV	Dado que o token corrente é SCONJ Quando este token for “como” E este token ocupar a primeira posição da sentença E for seguido por um token E esse token for VERB Então substituir a etiqueta do token corrente SCONJ para ADV	1	Lexicalizada	Língua geral	
Regra 35	SCONJ	ADV	Dado que o token corrente é SCONJ Quando este token for seguido por um token E esse token for “que” Então substituir a etiqueta do token corrente SCONJ para ADV	2	Lexicalizada	Língua geral	
Regra 36	VERB	ADV	Dado que o token corrente é VERB Quando este token for “fora” Então substituir a etiqueta do token corrente VERB para ADV	2	Lexicalizada	Língua geral	
Regra 37	ADJ	DET	Dado que o token corrente é ADJ Quando este token for precedido por um token E esse token for DET Então substituir a etiqueta do token corrente ADJ para DET	2	Não-lexicalizada	Língua geral	
Regra 38	ADP	DET	Dado que o token corrente é ADP Quando este token for precedido por um token e seguido por outro token E o token precedente for NOUN E o token seguinte for NOUN Então substituir a etiqueta do token corrente ADP para DET	2	Não-lexicalizada	Língua geral	
Regra 39	ADP	DET	Dado que o token corrente é ADP Quando este token for precedido por um token e seguido por outro token E o token precedente for VERB E o token seguinte for NOUN Então substituir a etiqueta do token corrente ADP para DET	2	Não-lexicalizada	Língua geral	
Regra 40	PRON	DET	Dado que o token corrente é PRON Quando este token for precedido por um ou dois tokens E pelo menos um desses tokens for PUNCT Então substituir a etiqueta do token corrente PRON para DET	2	Não-lexicalizada	Língua geral	
Regra 41	PRON	DET	Dado que o token corrente é PRON Quando este token for precedido por um ou dois tokens E pelo menos um desses tokens for DET Então substituir a etiqueta do token corrente PRON para DET	2	Não-lexicalizada	Língua geral	
Regra 42	SCONJ	DET	Dado que o token corrente é SCONJ Quando esse token for “que” E for seguido por um token	1	Lexicalizada	Língua geral	

			E esse token for “ganhos” Então substituir a etiqueta do token corrente SCONJ para DET				
Regra 43	SCONJ	DET	Dado que o token corrente é SCONJ Quando esse token for “que” E for seguido por um token E esse token for “tal” Então substituir a etiqueta do token corrente SCONJ para DET	1	Lexicalizada	Língua geral	
Regra 44	ADJ	INTJ	Dado que o token corrente é ADJ Quando este token for seguido por um token E esse token for PUNCT Então substituir a etiqueta do token corrente ADJ para INTJ	2	Não-lexicalizada	Língua geral	
Regra 45	ADV	INTJ	Dado que o token corrente é ADV Quando este token for “né” Então substituir a etiqueta do token corrente ADV para INTJ	1	Lexicalizada	Domínio	Coloquialismo
Regra 46	ADV	INTJ	Dado que o token corrente é ADV Quando este token for “amém” Então substituir a etiqueta do token corrente ADV para INTJ	1	Lexicalizada	Língua geral	
Regra 47	ADV	INTJ	Dado que o token corrente é ADV Quando este token for “ah” Então substituir a etiqueta do token corrente ADV para INTJ	1	Lexicalizada	Língua geral	
Regra 48	AUX	INTJ	Dado que o token corrente é AUX Quando este token for seguido por um token E esse token for PUNCT Então substituir a etiqueta do token corrente AUX para INTJ	1	Não-lexicalizada	Língua geral	
Regra 49	NOUN	INTJ	Dado que o token corrente é NOUN Quando este token for “blz” Então substituir a etiqueta do token corrente NOUN para INTJ	1	Lexicalizada	Domínio	Abreviação
Regra 50	NOUN	INTJ	Dado que o token corrente é NOUN Quando este token for “ops” Então substituir a etiqueta do token corrente NOUN para INTJ	1	Lexicalizada	Domínio	Coloquialismo
Regra 51	PRON	INTJ	Dado que o token corrente é PRON Quando este token for “ô” Então substituir a etiqueta do token corrente NOUN para INTJ	1	Lexicalizada	Língua geral	Coloquialismo
Regra 52	ADJ	NOUN	Dado que o token corrente não é NOUN Quando este token for “gasolina” Então substituir a etiqueta do token corrente para NOUN	1	Lexicalizada	Língua geral	
Regra 53	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for “b” E este token por precedido por um token E este token for “plano” Então substituir a etiqueta do token corrente ADJ para NOUN	1	Lexicalizada	Língua geral	
Regra 54	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for precedido por um token e seguido por outro token	1	Não-lexicalizada	Língua geral	

			E o token precedente for DET E o token seguinte não for NOUN Então substituir a etiqueta do token corrente ADJ para NOUN				
Regra 55	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for precedido por um token e seguido por outro token E o token precedente for VERB E o token seguinte não for ADP Então substituir a etiqueta do token corrente ADJ para NOUN	1	Não-lexicalizada	Língua geral	
Regra 56	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for "2ª", "3ª", "4ª", "5ª" ou "6ª" E este token for precedido por dois tokens E o segundo token precedente for ADP E o primeiro token precedente for DET Então substituir a etiqueta do token corrente ADJ para NOUN	1	Lexicalizada	Língua geral	
Regra 57	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for "acionista" ou "acionistas" Então substituir a etiqueta do token corrente ADJ para NOUN	1	Lexicalizada	Língua geral	
Regra 58	AD	NOUN	Dado que o token corrente é ADJ Quando este token for "estatal" ou "estatais" E este token for precedido por um token E este token não for NOUN Então substituir a etiqueta do token corrente ADJ para NOUN	1	Lexicalizada	Língua geral	
Regra 59	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for precedido por um token e seguido por outro token E o token precedente for DET E o token seguinte não for NOUN Então substituir a etiqueta do token corrente ADJ para NOUN	1	Não-lexicalizada	Língua geral	
Regra 60	ADJ	NOUN	Dado que os tokens correntes são respectivamente ADJ e NOUN Quando estes tokens forem precedidos por um token e seguido por outros dois tokens E o token precedente for DET E o primeiro token seguinte for DET ou CCONJ E o segundo token seguinte for ADJ Então inverter as etiquetas dos tokens correntes ADJ e NOUN para NOUN e ADJ, respectivamente.	1	Não-lexicalizada	Língua geral	
Regra 61	ADJ	NOUN	Dado que os tokens correntes são respectivamente NOUN e ADJ Quando estes tokens forem precedidos por um token e seguido por outros dois tokens E o token precedente for DET E o primeiro token seguinte for DET ou CCONJ E o segundo token seguinte for ADJ Então inverter as etiquetas dos tokens correntes NOUN e ADJ para ADJ e NOUN, respectivamente.	1	Não-lexicalizada	Língua geral	
Regra 62	PRON	NOUN	Dado que o token corrente é PRON Quando este token for "el" Quando este token for precedido por um ou dois tokens E pelo menos um desses tokens for VERB Então substituir a etiqueta do token corrente PRON para NOUN	1	Lexicalizada	Domínio	Truncamento

Regra 63	SYM	NOUN	Dado que o token corrente é SYM Quando este token for “abs” Então substituir a etiqueta do token corrente SYM para NOUN	1	Lexicalizada	Domínio	Abreviação
Regra 64	VERB	NOUN	Dado que o token corrente é VERB Quando este token for “escolha” E for seguido por um token E esse token for “bela” etiquetado como PROPN Então substituir a etiqueta do token corrente VERB para NOUN e substituir a etiqueta do token “bela” para ADJ.	1	Lexicalizada	Língua geral	
Regra 65	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por um token e seguido por outro token E o token precedente for PROPN E o token seguinte for “:” etiquetado como PUNCT Então substituir a etiqueta do token corrente VERB para NOUN	1	Lexicalizada	Língua geral	
Regra 66	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por um token E o token precedente não for PRON Então substituir a etiqueta do token corrente VERB para NOUN	1	Não-lexicalizada	Língua geral	
Regra 67	VERB	NOUN	Dado que o token corrente é VERB Quando este token for “desculpa” E este token for precedido por um token E o token precedente não for PRON Então substituir a etiqueta do token corrente VERB para NOUN	1	Lexicalizada	Língua geral	
Regra 68	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por um token E o token precedente for “que” Então substituir a etiqueta do token corrente VERB para NOUN	1	Lexicalizada	Língua geral	
Regra 69	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por dois tokens e seguido por outro token E o segundo token precedente for NOUN E o primeiro token precedente for CCONJ E o token seguinte for DET ou ADP Então substituir a etiqueta do token corrente VERB para NOUN	1	Não-lexicalizada	Língua geral	
Regra 70	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por dois tokens E o segundo token precedente for ADP E o primeiro token precedente for PROPN Então substituir a etiqueta do token corrente VERB para NOUN	1	Não-lexicalizada	Língua geral	
Regra 71	ADV	PRON	Dado que o token corrente é ADV Quando este token for “algo” Então substituir a etiqueta do token corrente ADV para PRON	1	Lexicalizada	Língua geral	
Regra 72	ADV	PRON	Dado que o token corrente é ADV Quando este token for precedido por um token E este token for “que” Então substituir a etiqueta do token corrente ADV para PRON	1	Lexicalizada	Língua geral	

Regra 73	DET	PRON	Dado que o token corrente é DET Quando este token for “o” E este toke for seguido por um token E este token for “q” Então substituir a etiqueta do token corrente DET para PRON	2	Lexicalizada	Língua geral	
Regra 74	DET	PRON	Dado que o token corrente é DET Quando este token for seguido por um token E este toke for NOUN Então substituir a etiqueta do token corrente DET para PRON	1	Não-lexicalizada	Língua geral	
Regra 75	NOUN	PRON	Dado que o token corrente é NOUN Quando este token for “vcs” Então substituir a etiqueta do token corrente NOUN para PRON	2	Lexicalizada	Domínio	Abreviação
Regra 76	NOUN	PRON	Dado que o token corrente é NOUN Quando este token for “mim” Então substituir a etiqueta do token corrente NOUN para PRON	1	Lexicalizada	Língua geral	Abreviação
Regra 77	ADP	SCONJ	Dado que o token corrente é ADP Quando este token for precedido por três tokens e seguido por três outros tokens E pelo menos um dos tokens precedentes forem ADV ou VERB E pelo menos um dos tokens seguintes forem VERB ou AUX Então substituir a etiqueta do token corrente ADP para SCONJ	1	Não-lexicalizada	Língua geral	
Regra 78	ADV	SCONJ	Dado que o token corrente é ADV Quando este token for precedido por dois tokens E pelo menos um dos tokens precedentes for VERB Então substituir a etiqueta do token corrente ADV para SCONJ	2	Não-lexicalizada	Língua geral	
Regra 79	PRON	SCONJ	Dado que o token corrente é PRON Quando este token for seguido por três tokens E pelo menos um dos tokens seguintes for VERB Então substituir a etiqueta do token corrente PRON para SCONJ	3	Não-lexicalizada	Língua geral	Simbolo
Regra 80	NOUN	SYM	Dado que o token corrente é NOUN Quando este token for “x” Então substituir a etiqueta do token corrente NOUN para SYM	1	Lexicalizada	Domínio	Truncamento
Regra 81	ADJ	VERB	Dado que o token corrente é ADJ Quando este token for “regist” Então substituir a etiqueta do token corrente ADJ para VERB	1	Lexicalizada	Domínio	Truncamento
Regra 82	ADJ	VERB	Dado que o token corrente é ADJ Quando este token for “fazen” Então substituir a etiqueta do token corrente ADJ para VERB	1	Lexicalizada	Domínio	
Regra 83	ADP	VERB	Dado que o token corrente é ADP Quando este token for precedido por um token E o token precedente for ADV Então substituir a etiqueta do token corrente ADP para VERB	1	Não-lexicalizada	Língua geral	
Regra 84	ADP	VERB	Dado que o token corrente é ADP Quando este token for seguido por um token	1	Não-lexicalizada	Língua geral	

			E o token seguinte for CCONJ Então substituir a etiqueta do token corrente ADP para VERB				
Regra 85	ADP	VERB	Dado que o token corrente é ADP Quando este token for “há” Então substituir a etiqueta do token corrente ADP para VERB	1	Lexicalizada	Língua geral	
Regra 86	AUX	VERB	Dado que o token corrente é AUX Quando este token for precedido por três tokens E nenhum dos tokens precedentes forem VERB Então substituir a etiqueta do token corrente AUX para VERB	4	Não-lexicalizada	Língua geral	
Regra 87	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for “monitoro” Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral	
Regra 88	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for “lembra” Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral	
Regra 89	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for “renova” Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral	
Regra 90	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for “trabalho” E este token for precedido por um token E este token não for DET Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral	
Regra 91	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for “venda” E este token for seguido por dois tokens E estes tokens serem “de” e “novo”, respectivamente Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral	
Regra 92	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for “preço” Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral	
Regra 93	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for “partir” Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral	

APÊNDICE 3 – REGRAS DE PÓS-EDIÇÃO DE POS TAGGING REFINADAS

Regra	DE	PARA	Descrição	Qt.	Tipo de regra	Linguagem	Categoria de refinamento	Observação
Regra 1	ADV	ADJ	Dado que o token corrente é ADV Quando este token for a palavra “certo” Então substituir a etiqueta do token corrente ADV para ADJ	1	Lexicalizada	Língua geral		
Regra 2	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for precedido por dois tokens E esses tokens são respectivamente DET e NOUN Então substituir a etiqueta do token corrente NOUN para ADJ	2	Não-lexicalizada	Língua geral		
Regra 3	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for precedido por um token e seguido por outro token E o token precedente for ADP E o token seguinte for NOUN Então substituir a etiqueta do token corrente NOUN para ADJ	1	Não-lexicalizada	Língua geral		
Regra 4	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for a palavra “log”, “casado”, “melho” ou “voláteis” E ser precedido por um token E esse token não é “LLX” Então substituir a etiqueta do token corrente NOUN para ADJ	4	Lexicalizada	Língua geral e domínio	Aglutinação e Exceção	Antigas regras 6,7,9 e 10
Regra 5	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for a palavra “coitada” E este token ocupar a primeira posição da sentença E este token for seguido de um token ADP Então substituir a etiqueta do token corrente NOUN para ADJ	1	Lexicalizada	Língua geral		
Regra 6	NOUN	ADJ	Dado que o token corrente é NOUN Quando este token for precedido por dois tokens E esses tokens são respectivamente PUNCT e ADJ E o token PUNCT for “/” Então substituir a etiqueta do token corrente NOUN para ADJ	1	Não lexicalizada	Língua geral		
Regra 7	NOUN	ADJ	Dado que o token corrente é VERB Quando este token tiver os últimos 3 caracteres com a sequência -ado, ou -ados, ou -ido, ou -idos E esses token for diretamente precedido de NOUN Então substituir a etiqueta do token corrente VERB para ADJ	7	Não-lexicalizada	Língua geral	Aglutinação e generalização	Antigas regras 11, 12, 14 e 15
Regra 8	NOUN	ADJ	Dado que o token corrente é VERB Quando este token for a palavra “lindo” Então substituir a etiqueta do token corrente VERB para ADJ	1	Lexicalizada	Língua geral		
Regra 9	ADV	ADP	Dado que o token corrente é ADV Quando este token for precedido por um token E esse token for “assim” Então substituir a etiqueta do token corrente ADV para ADP	1	Lexicalizada	Língua geral		

Regra 10	DET	ADP	Dado que o token corrente é DET Quando este token for "a" E esse token for precedido por um token E esse token for VERB Então substituir a etiqueta do token corrente DET para ADP	1	Lexicalizada	Língua geral	Lexicalização	A palavra "a" pode desempenhar a função de DET e ADP, portanto, é mais recomendável especificar a palavra do que generalizar a classe toda e gerar possíveis falsos positivos
Regra 11	NOUN	ADP	Dado que o token corrente não é ADP Quando este token for precedido por um token E esse token for “market” Então substituir a etiqueta do token corrente para ADP	1	Lexicalizada	Domínio		
Regra 12	NOUN	ADP	Dado que o token corrente não é ADP Quando este token for “d” Então substituir a etiqueta do token corrente para ADP	1	Lexicalizada	Língua geral		
Regra 13	SCONJ	ADP	Dado que o token corrente é SCONJ Quando este token for a, ante, até, após, com, contra, de, desde, em, entre, para, perante, por, sem, sob ou sobre Então substituir a etiqueta do token corrente SCONJ para ADP.	7	Não-lexicalizada	Língua geral	Aglutinação	Antiga regra 20, 21, 22, 23
Regra 14	ADP	ADV	Dado que o token corrente é ADP Quando este token for seguido por um token E esse token for “vc” Então substituir a etiqueta do token corrente ADP para ADV	1	Lexicalizada	Língua geral		
Regra 15	ADP	ADV	Dado que o token corrente é ADP Quando este token for seguido por um token E esse token for “mesmo” Então substituir a etiqueta do token corrente ADP para ADV	1	Lexicalizada	Língua geral		
Regra 16	NOUN	ADV	Dado que o token corrente é NOUN Quando este token for “tb”, "menos", "avante", "cm", ou "msm" Então substituir a etiqueta do token corrente NOUN para ADV	5	Lexicalizada	Domínio	Aglutinação	Antiga regra 26, 27, 28, 29, 30
Regra 17	NOUN	ADV	Dado que o token corrente é NOUN Quando este token for "baixo" E esse token for precedido por um token E esse token for ADP Então substituir a etiqueta do token corrente NOUN para ADV	2	Lexicalizada	Língua geral	Lexicalização	A regra não era clara se tratava apenas da palavra "baixo" ou de toda classe NOUN, portanto, tornou-se lexicalizada a fim de evitar falsos positivos, pois não é recomendável afirmar que todo NOUN precedido de ADP é ADV.
Regra 18	PRON	ADV	Dado que o token corrente é PRON Quando este token for precedido por um token E esse token for “nojo” Então substituir a etiqueta do token corrente PRON para ADV	1	Lexicalizada	Língua geral		
Regra 19	PRON	ADV	Dado que o token corrente é PRON Quando este token for “pouco” E estiver precedido de um token E esse token for VERB Então substituir a etiqueta do token corrente PRON para ADV	1	Lexicalizada	Língua geral		

Regra 20	SCONJ	ADV	Dado que o token corrente é SCONJ Quando este token for “como” E este token ocupar a primeira posição da sentença E for seguido por um token E esse token for VERB Então substituir a etiqueta do token corrente SCONJ para ADV	1	Lexicalizada	Língua geral		
Regra 21	SCONJ	ADV	Dado que o token corrente é SCONJ Quando este token for seguido por um token E esse token for “que” Então substituir a etiqueta do token corrente SCONJ para ADV	2	Lexicalizada	Língua geral		
Regra 22	VERB	ADV	Dado que o token corrente é VERB Quando este token for “fora” Então substituir a etiqueta do token corrente VERB para ADV	2	Lexicalizada	Língua geral		
Regra 23	ADP	DET	Dado que o token corrente é ADP Quando este não for "a" E esste token for precedido por um token e seguido por outro token E o token precedente for VERB E o token seguinte for NOUN Então substituir a etiqueta do token corrente ADP para DET	2	Não-lexicalizada	Língua geral	Exceção	A palavra "a" pode exercer função de DET e ADP, a depender do contexto, portanto é recomendável fazer a exceção para reduzir a quantidade de falsos positivos.
Regra 24	PRON	DET	Dado que o token corrente é PRON Quando este token for precedido por um ou dois tokens E pelo menos um desses tokens for PUNCT Então substituir a etiqueta do token corrente PRON para DET	2	Não-lexicalizada	Língua geral		
Regra 25	PRON	DET	Dado que o token corrente é PRON Quando este token for precedido por um ou dois tokens E pelo menos um desses tokens for DET Então substituir a etiqueta do token corrente PRON para DET	2	Não-lexicalizada	Língua geral		
Regra 26	SCONJ	DET	Dado que o token corrente é SCONJ Quando esse token for “que” E for seguido por um token E esse token for “ganhos” Então substituir a etiqueta do token corrente SCONJ para DET	1	Lexicalizada	Língua geral		
Regra 27	SCONJ	DET	Dado que o token corrente é SCONJ Quando esse token for “que” E for seguido por um token E esse token for “tal” Então substituir a etiqueta do token corrente SCONJ para DET	1	Lexicalizada	Língua geral		
Regra 28	ADJ	INTJ	Dado que o token corrente é ADJ Quando este token for seguido por um token E esse token for "!" etiquetado como PUNCT Então substituir a etiqueta do token corrente ADJ para INTJ	2	Não-lexicalizada	Língua geral	Lexicalização	É preciso especificar qual pontuação se refere, pois é comum ADJ estarem seguidos de PUNCT, como vírgula, ponto final, interrogação, etc. As INTJ ocorrem comumente com o ponto de exclamação

Regra 29	ADV	INTJ	Dado que o token corrente é ADV, NOUN ou PRON Quando este token for “né”, "ah", "blz", "ops", "ô", Então substituir a etiqueta do token corrente ADV para INTJ	1	Lexicalizada	Domínio	Aglutinação	Antiga regra 45, 46, 47, 48, 49, 50, 51
Regra 30	AUX	INTJ	Dado que o token corrente é AUX Quando este token for seguido por um token E esse token for PUNCT Então substituir a etiqueta do token corrente AUX para INTJ	1	Não-lexicalizada	Língua geral		
Regra 31	ADJ	NOUN	Dado que o token corrente não é NOUN Quando este token for “gasolina” Então substituir a etiqueta do token corrente para NOUN	1	Lexicalizada	Língua geral		
Regra 32	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for “b” E este token por precedido por um token E este token for “plano” Então substituir a etiqueta do token corrente ADJ para NOUN	1	Lexicalizada	Língua geral		
Regra 33	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for precedido por um token e seguido por outro token E o token precedente for DET E o token seguinte não for NOUN Então substituir a etiqueta do token corrente ADJ para NOUN	1	Não-lexicalizada	Língua geral		
Regra 34	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for precedido por um token e seguido por outro token E o token precedente for VERB E o token seguinte não for ADP Então substituir a etiqueta do token corrente ADJ para NOUN	1	Não-lexicalizada	Língua geral		
Regra 35	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for “2ª”, “3ª”, “4ª”, “5ª” ou “6ª” E este token for precedido por dois tokens E o segundo token precedente for ADP E o primeiro token precedente for DET Então substituir a etiqueta do token corrente ADJ para NOUN	1	Lexicalizada	Língua geral		
Regra 36	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for “acionista” ou “acionistas” Então substituir a etiqueta do token corrente ADJ para NOUN	1	Lexicalizada	Língua geral		
Regra 37	AD	NOUN	Dado que o token corrente é ADJ Quando este token for “estatal” ou “estatais” E este token for precedido por um token E este token não for NOUN Então substituir a etiqueta do token corrente ADJ para NOUN	1	Lexicalizada	Língua geral		
Regra 38	ADJ	NOUN	Dado que o token corrente é ADJ Quando este token for precedido por um token e seguido por outro token E o token precedente for DET E o token seguinte não for NOUN Então substituir a etiqueta do token corrente ADJ para NOUN	1	Não-lexicalizada	Língua geral		

Regra 39	ADJ	NOUN	Dado que os tokens correntes são respectivamente ADJ e NOUN Quando estes tokens forem precedidos por um token e seguido por outros dois tokens E o token precedente for DET E o primeiro token seguinte for DET ou CCONJ E o segundo token seguinte for ADJ Então inverter as etiquetas dos tokens correntes ADJ e NOUN para NOUN e ADJ, respectivamente.	1	Não-lexicalizada	Língua geral		
Regra 40	PRON	NOUN	Dado que o token corrente é PRON Quando este token for “el” Quando este token for precedido por um ou dois tokens E pelo menos um desses tokens for VERB Então substituir a etiqueta do token corrente PRON para NOUN	1	Lexicalizada	Domínio		
Regra 41	SYM	NOUN	Dado que o token corrente é SYM Quando este token for “abs” Então substituir a etiqueta do token corrente SYM para NOUN	1	Lexicalizada	Domínio		
Regra 42	VERB	NOUN	Dado que o token corrente é VERB Quando este token for “escolha” E for seguido por um token E esse token for “bela” etiquetado como PROPN Então substituir a etiqueta do token corrente VERB para NOUN e substituir a etiqueta do token “bela” para ADJ.	1	Lexicalizada	Língua geral		
Regra 43	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por um token e seguido por outro token E o token precedente for PROPN E o token seguinte for “:” etiquetado como PUNCT Então substituir a etiqueta do token corrente VERB para NOUN	1	Lexicalizada	Língua geral		
Regra 44	VERB	NOUN	Dado que o token corrente é VERB Quando este token for “desculpa” E este token for precedido por um token E o token precedente não for PRON Então substituir a etiqueta do token corrente VERB para NOUN	1	Lexicalizada	Língua geral		
Regra 45	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por dois tokens e seguido por outro token E o segundo token precedente for NOUN E o primeiro token precedente for CCONJ E o token seguinte for DET ou ADP Então substituir a etiqueta do token corrente VERB para NOUN	1	Não-lexicalizada	Língua geral		
Regra 46	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por dois tokens E o segundo token precedente for ADP E o primeiro token precedente for PROPN Então substituir a etiqueta do token corrente VERB para NOUN	1	Não-lexicalizada	Língua geral		
Regra 47	ADV	PRON	Dado que o token corrente é ADV Quando este token for “algo” Então substituir a etiqueta do token corrente ADV para PRON	1	Lexicalizada	Língua geral		

Regra 48	ADV	PRON	Dado que o token corrente é ADV Quando este token for "tal" E esse token for precedido por um token E este token for “que” Então substituir a etiqueta do token corrente ADV para PRON	1	Lexicalizada	Língua geral	Lexicalização	A regra é muito ampla e pode gerar falsos positivos, por isso é preciso especificar os itens lexicais para torná-la mais pontual
Regra 49	DET	PRON	Dado que o token corrente é DET Quando este token for “o” E este toke for seguido por um token E este token for “q” Então substituir a etiqueta do token corrente DET para PRON	2	Lexicalizada	Língua geral		
Regra 50	NOUN	PRON	Dado que o token corrente é NOUN Quando este token for “vcs” Então substituir a etiqueta do token corrente NOUN para PRON	2	Lexicalizada	Domínio		
Regra 51	NOUN	PRON	Dado que o token corrente é NOUN Quando este token for “mim” Então substituir a etiqueta do token corrente NOUN para PRON	1	Lexicalizada	Língua geral		
Regra 52	ADP	SCONJ	Dado que o token corrente é ADP Quando este token for precedido por três tokens e seguido por três outros tokens E pelo menos um dos tokens precedentes forem ADV ou VERB E pelo menos um dos tokens seguintes forem VERB ou AUX Então substituir a etiqueta do token corrente ADP para SCONJ	1	Não-lexicalizada	Língua geral		
Regra 53	ADV	SCONJ	Dado que o token corrente é ADV Quando este token for precedido por dois tokens E pelo menos um dos tokens precedentes for VERB Então substituir a etiqueta do token corrente ADV para SCONJ	2	Não-lexicalizada	Língua geral		
Regra 54	PRON	SCONJ	Dado que o token corrente é PRON Quando este token for seguido por três tokens E pelo menos um dos tokens seguintes for VERB Então substituir a etiqueta do token corrente PRON para SCONJ	3	Não-lexicalizada	Língua geral		
Regra 55	NOUN	SYM	Dado que o token corrente é NOUN Quando este token for “x” Então substituir a etiqueta do token corrente NOUN para SYM	1	Lexicalizada	Domínio		
Regra 56	ADJ	VERB	Dado que o token corrente é ADJ Quando este token for “regist” Então substituir a etiqueta do token corrente ADJ para VERB	1	Lexicalizada	Domínio		
Regra 57	ADJ	VERB	Dado que o token corrente é ADJ Quando este token for “fazen” Então substituir a etiqueta do token corrente ADJ para VERB	1	Lexicalizada	Domínio		
Regra 58	ADP	VERB	Dado que o token corrente é ADP Quando este token for "para" E o token precedente for ADV Então substituir a etiqueta do token corrente ADP para VERB	1	Não-lexicalizada	Língua geral		

Regra 59	ADP	VERB	Dado que o token corrente é ADP Quando este token for seguido por um token E o token seguinte for CCONJ Então substituir a etiqueta do token corrente ADP para VERB	1	Não-lexicalizada	Língua geral		
Regra 60	ADP	VERB	Dado que o token corrente é ADP Quando este token for "há" Então substituir a etiqueta do token corrente ADP para VERB	1	Lexicalizada	Língua geral		
Regra 61	AUX	VERB	Dado que o token corrente é AUX Quando este token for precedido por três tokens E nenhum dos tokens precedentes forem VERB Então substituir a etiqueta do token corrente AUX para VERB	4	Não-lexicalizada	Língua geral		
Regra 62	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for "monitoro" Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral		
Regra 63	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for "lembra" Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral		
Regra 64	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for "renova" Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral		
Regra 65	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for "trabalho" E este token for precedido por um token E este token não for DET Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral		
Regra 66	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for "venda" E este token for seguido por dois tokens E estes tokens serem "de" e "novo", respectivamente Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral		
Regra 67	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for "preço" Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral		
Regra 68	NOUN	VERB	Dado que o token corrente é NOUN Quando este token for "partir" Então substituir a etiqueta do token corrente NOUN para VERB	1	Lexicalizada	Língua geral		

APÊNDICE 4 – REGRAS EXCLUIDAS APÓS REFINAMENTO

Regra	DE	PARA	Descrição	QT	Tipo de regra	Linguagem	Categoria de refinamento	Observação
Regra 37	ADJ	DET	Dado que o token corrente é ADJ Quando este token for precedido por um token E esse token for DET Então substituir a etiqueta do token corrente ADJ para DET	2	Não-lexicalizada	Língua geral	Exclusão	A sequência DET + ADJ é comum na língua portuguesa, por isso não é adequado dizer que todo ADJ antecedido por DET é DET também.
Regra 38	ADP	DET	Dado que o token corrente é ADP Quando este token for precedido por um token e seguido por outro token E o token precedente for NOUN E o token seguinte for NOUN Então substituir a etiqueta do token corrente ADP para DET	2	Não-lexicalizada	Língua geral	Exclusão	A sequência NOUN + ADP + NOUN é comum na língua portuguesa. Os sintagmas preposicionais carregam essa estrutura, por isso não é adequado dizer que todo ADP nessa estrutura é DET.
Regra 61	ADJ	NOUN	Dado que os tokens correntes são respectivamente NOUN e ADJ Quando estes tokens forem precedidos por um token e seguido por outros dois tokens E o token precedente for DET E o primeiro token seguinte for DET ou CCONJ E o segundo token seguinte for ADJ Então inverter as etiquetas dos tokens correntes NOUN e ADJ para ADJ e NOUN, respectivamente.	1	Não-lexicalizada	Língua geral	Exclusão	A regra 60 e 61 causam um looping, pois a alteração realizada por uma regra ativa a identificação da outra, e vice e versa. Portanto é preciso excluir uma delas.
Regra 66	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por um token E o token precedente não for PRON Então substituir a etiqueta do token corrente VERB para NOUN	1	Não-lexicalizada	Língua geral	Exclusão	Várias etiquetas são comuns de ocorrer antes de VERB, como ADV, NOUN, ADJ, PRON, ADP, CCONJ, SCONJ. Portanto, não é recomendável alterar para NOUN todo VERB antecedido por essas etiquetas
Regra 68	VERB	NOUN	Dado que o token corrente é VERB Quando este token for precedido por um token E o token precedente for “que” Então substituir a etiqueta do token corrente VERB para NOUN	1	Lexicalizada	Língua geral	Exclusão	É comum a palavra "que" preceder verbos, principalmente em orações subordinadas, portanto, esse é um padrão que possivelmente gerará muitos falsos positivos
Regra 74	DET	PRON	Dado que o token corrente é DET Quando este token for seguido por um token E este toke for NOUN Então substituir a etiqueta do token corrente DET para PRON	1	Não-lexicalizada	Língua geral	Exclusão	A sequência DET + NOUN é muito comum na língua portuguesa, portanto, não é recomendável sempre que ocorrer essa sequência alterar para DET + PRON, pois gerará muitos falsos positivos