UDFest-BR 2022

Universal Dependencies Brazilian Festival

Proceedings of the Conference, Vol. 1

March 21, 2022

About the workshop

The Universal Dependencies Brazilian Festival (UDFest-BR) took place with the 15th edition of the International Conference on the Computational Processing of Portuguese (PROPOR 2022). It is a forum in which researchers involved with the study and application of the Universal Dependencies (UD) model, along with its adaptation to Portuguese, can meet and discuss best practices and strategies, as well as problems and solutions related to this topic.

UD is a cross-language international model for grammatical annotation (morphosyntactic tags, lexical/morphosyntactic features and syntactic dependencies), which results from the efforts of an open community of hundreds of researchers. Building upon a common framework, the model captures idiosyncrasies and similarities between different languages, aiming at fostering contrastive linguistic studies and the development of multilingual technologies for Natural Language Processing (NLP).

In recent years, this model has drawn considerable attention from the NLP community. This growing interest can be owed mainly to the fact that UD allows for the development of fairly precise taggers and parsers to a number of languages, based on the exploitation of syntactically annotated corpora (treebanks) and some Machine Learning algorithms. Currently, we find around 200 UD annotated corpora available in more than 100 languages.

Although there can be found some resources available in Portuguese, computational linguistic researches involving UD and this language are still incipient. As such, the existence of a workshop dedicated to this topic certainly helps to fill in this gap, giving more visibility to current efforts in this area as well as fostering new initiatives and collaborations.

This edition of the workshop includes six papers of renewed researchers in the area for the Portuguese language. There are theoretical discussions, reports on corpus annotation and the related challenges, and software for UD-related processing. The papers and their authors are:

- *Anotação de textos não canônicos: um estudo exploratório de Grande sertão: veredas pelas dependências universais*
  André V. L. Coneglian, Ana Luísa A. R. Guimarães, Thiago C. Ferreira, Adriana S. Pagano

- *Polishing the gold – how much revision do we need in treebanks?*
  Elvis de Souza, Cláudia Freitas

- *Que simples que nada: a anotação da palavra que em córpus de UD*
  Magali Duran, Heloísa Oliveira, Clarissa Scandarolli

- *Shallow parsing of Portuguese texts annotated under Universal Dependencies*
  Guilherme M. Oliveira, Paulo Berlanga Neto, Evandro Ruiz

- *Still on arguments and adjuncts: the status of the indirect object and the adverbial adjunct relations in Universal Dependencies for Portuguese*
  Elvis de Souza, Cláudia Freitas

- *UDConcord: A Concordancer for Universal Dependencies Treebanks*
  Lucas G. M. Miranda, Thiago A. S. Pardo

Organizing committee

This workshop is promoted by the POeTiSA project, which is a long-term initiative that aims at growing syntax-based resources and developing related tools and applications for Brazilian Portuguese language, looking to achieve world state-of-the-art results in this area. The project is part of the Natural Language Processing initiative (NLP2) of the Center for Artificial Intelligence (C4AI) of the University of São Paulo, sponsored by IBM and FAPESP.

The following researchers organized this edition of the workshop:

Thiago Alexandre Salgueiro Pardo
University of São Paulo (USP)
São Carlos/SP, Brazil

Ariani Di Felippo
Federal University of São Carlos (UFSCar)
São Carlos/SP, Brazil

Norton Trevisan Roman
University of São Paulo (USP)
São Paulo/SP, Brazil

Program committee

Adriana Silvina Pagano (Brazil)
Cláudia Freitas (Brazil)
Daniel Zeman (Czech Republic)
Joakim Nivre (Sweden)
Jorge Baptista (Portugal)
Leonardo Zilio (UK)
Lucelene Lopes (USA)
Magali Sanches Duran (Brazil)
Maria das Graças Volpe Nunes (Brazil)
Valeria de Paiva (USA)

Anotação de textos não canônicos: um estudo exploratório de *Grande sertão: veredas* pelas dependências universais

André V. L. Coneglian[1][0000-0003-1726-8890], Ana Luisa A. R. Guimarães[1][0000-0001-7278-6856], Thiago Castro Ferreira[1] [0000-0003-0200-3646], Adriana S. Pagano[1][0000-0002-3150-3503]
[1] Universidade Federal de Minas Gerais, Belo Horizonte MG 31270-901, BRA
coneglian@ufmg.br; alarp@ufmg.br; thiagocf05@ufmg.br; apagano@ufmg.br

Abstract. This paper reports on an exploratory study of a sample of 175 sentences retrieved from the renowned Brazilian novel *Grande Sertão: Veredas* [Portuguese for *Great Backlands: Paths*; English translation: *The devil to pay in the backlands*], which were annotated for POS and syntactic relations following the Universal Dependencies guidelines. The study aimed to explore the feasibility of annotating non-canonical text to create treebanks for Brazilian Portuguese. We computed accuracy and precision of the model in order to verify categories annotated more and less successfully. The results show the model performed slightly better for POS than dependency relations and pointed out categories with higher demand for manual revision as being those related to orality phenomena represented by Guimarães Rosa in his novel. The study shows the potential of annotating non-canonical text to enhance existing models with categories less represented in the treebanks.

Keywords: Universal Dependencies, non-canonical text, Brazilian Portuguese.

1 Introdução

A seleção de textos para fins de anotação e treinamento de modelos visando o Processamento de Língua Natural (PLN) tem privilegiado um conjunto limitado de tipos, dentre os quais se destaca o texto jornalístico com expressiva presença nos *treebanks* existentes nas distintas línguas (Silveira et al., 2014; Plank, 2016; Zeldes & Simonson, 2016; Bamman, 2020). Textos técnicos de domínios específicos, como biomedicina, também estão representados nos *corpora* disponíveis e, juntamente com os jornalísticos, constituem “textos canônicos” em PLN. Por outro lado, os chamados textos "não canônicos" vêm se fazendo cada vez mais presentes, e, geralmente, caracterizam-se como de difícil anotação tanto em termos de padrões gramaticais quanto em termos de limitações que esquemas de anotação apresentam para lidar com eles (Hirschmann et al., 2007). São considerados textos "não canônicos" aqueles que são extraídos das chamadas mídias sociais, como *Twitter*, fóruns e *blogs*, bem como narrativas clínicas, textos produzidos por aprendizes de língua e textos literários clássicos ou contemporâneos.

Em português brasileiro, língua considerada com recursos insuficientes para o desenvolvimento de aplicações em PLN, os *treebanks* existentes constituem-se de

textos jornalísticos e, mais recentemente, de textos de mídias sociais, bem como de narrativas clínicas. Não existe ainda um *treebank* de texto literário e, muito menos, de texto literário com uso não convencional da linguagem.

Este estudo apresenta uma anotação exploratória de uma amostra de um texto literário não convencional, o romance *Grande sertão: Veredas,* de João Guimarães Rosa, publicado em 1959, no Brasil, e caracterizado pela crítica literária como inovador tanto no léxico quanto na gramática. Enquanto proposta de uso criativo da linguagem, o romance rosiano também é considerado "não canônico" no escopo do PLN. O objetivo central é explorar a viabilidade de se anotar textos não canônicos, de natureza literária, para desenvolver *treebanks* em português brasiliero pautados pelas diretrizes do projeto Universal Dependencies.

A anotação das relações de dependência que se faz neste estudo vai na direção de explorar as fronteiras entre uma sintaxe da sentença e uma sintaxe do discurso (Kaltenböck et al., 2011; Heine et al., 2013; Heine et al., 2017), uma vez que a interpretação da escrita rosiana desafia as fronteiras desses dois domínios. Assim, para casos limítrofes em que o analista tem de decidir entre uma ou outra função, as decisões de anotação são tomadas considerando-se o co-texto da sentença analisada, isto é, as sentenças anteriores, não apenas o que se apresenta na sentença analisada.

O artigo está organizado em 5 seções, além desta Introdução. Na Seção 2, discute-se o estatuto da canonicidade dos textos em PLN, destacando-se as singularidades de *Grande sertão: Veredas*. Na Seção 3, apresenta-se a Metodologia deste estudo. A Seção 4 apresenta os resultados da anotação, bem como uma caracterização linguística da amostra analisada, apontando-se as categorias nas quais o modelo de anotação automática teve boa performance e aquelas que desafiam o modelo, por estarem menos representadas. Na seção 5, os resultados são interpretados e discutidos com base nas considerações sobre o caráter não canônico do texto analisado neste estudo. A seção 6 recupera e sintetiza os principais argumentos desenvolvidos no estudo.

2 A linguagem de *Grande sertão: Veredas*

2.1 O estatuto "não canônico" dos textos em PLN

No campo do PLN, a canonicidade de um tipo de texto parece ser determinada pelo desvio observado em relação aos padrões mais convencionais de um sistema linguístico ou bem pela dificuldade que seu processamento gera aos modelos existentes (Hirschman et al., 2007). Fenômenos gramaticais geralmente rotulados como fora de um padrão representam casos de fluidez categorial e de multifuncionalidade dos itens linguísticos (Neves, 2012). Sua não canonicidade resolve-se, em última instância, em uma questão de como esses fatos são acomodados teoricamente.

A avaliação que geralmente se faz da linguagem rosiana, especialmente em *Grande sertão: Veredas*, aponta o léxico como requintado e a sintaxe como dificultosa. O léxico de *Grande sertão: Veredas* é resultado de um trabalho criativo, que, mesclando raízes e morfemas de diferentes línguas, resulta em inovações lexicais e fraseológicas, muito bem documentadas lexicograficamente em Martins (2001). O aproveitamento para os *treebanks* é evidente nesse ponto, destacando-se, principalmente, a ampliação lexical dos bancos de dados, por meio da anotação do sistema de classes de palavras, com o auxílio de obras lexicográficas (Martins, 2001). No que diz respeito à sintaxe, a

complexidade do texto de *GS:V* parece ser decorrente de "inversões e elipses" e de "construções carregadas de ênfase" (Martins, 2001, p. XI). Ocorre que a sintaxe de Guimarães Rosa nunca viola as possibilidades sistêmicas da sintaxe portuguesa; antes, o autor encontra espaços de manobra que lhe permitem explorar as estratégias construcionais do português.

2.2 Aspectos lexicogramaticais da obra

Para entender a constituição linguística de *Grande sertão: Veredas* é necessário recorrer à própria configuração da obra. O romance se configura com uma conversa monológica entre o narrador-personagem, Riobaldo, e seu interlocutor, interpelado como "senhor", a quem conta histórias da sua época de jagunço no sertão mineiro, como se mostra em (1). Esse ponto de partida é importante para a consideração da linguagem na obra, porque o que Rosa faz é construir a "linguagem falada" pelo narrador num projeto literário de explorar as possibilidades lexicais e gramaticais em múltiplos registros de um português plurilíngue (Rocha, 2021), como se vê em (2).

(1) O senhor aprova? Me declare tudo, franco — é alta mercê que me faz: e pedir posso, encarecido. (GS:V)
(2) Hem? Hem? Ah. Figuração minha, de pior pra trás, as certas lembranças. (GS:V)

O fato de Rosa configurar o seu romance como se fosse o registro de um relato feito por um jagunço do sertão mineiro implica a incorporação de diversos fenômenos da modalidade falada da língua à modalidade escrita. Assim, as omissões, inversões e elisões que Martins (2001) caracterizou como aspectos dificultosos da sintaxe rosiana são, na verdade e de fato, mecanismos do funcionamento natural da língua falada. Esses mecanismos estão muito bem documentados para o português brasileiro (Castilho, 2002a, 2002b; Ilari, 2002; Castilho e Basílio, 2002; Kato, 2002; Koch, 2002; Neves, 1999; Abaurre e Rodrigues, 2002). Dizem Tarallo, Kato et al (1989, p. 25 – destaque original) que "uma primeira tomada de contato de um *corpus* natural de linguagem oral leva-nos a perceber a quantidade de emissões que, em relação a uma estrutura canônica do tipo *sujeito* + *predicado* (...)" apresentam-se, em alguma medida, fora dessa estrutura. O trecho em (3) ilustra a caracterização de Tarallo, Kato et al (1989).

(3) Se a gente — conforme compadre meu Quelemém é quem diz — se a gente torna a encarnar renovado, eu cismo até que inimigo de morte pode vir como filho do inimigo. (GS:V)

Em (3), à semelhança do que ocorre na língua falada, o narrador começa a formular seu enunciado com "Se a gente", mas logo o interrompe com a inclusão de uma estrutura adverbial, para depois retomar a formulação do seu enunciado com uma reiteração de "se a gente". Uma outra característica da língua falada presente no romance é a topicalização e deslocamento de constituintes para a posição inicial da sentença, como ilustrado por (4) e (5).

(4) Dono dele nem sei quem for. (GS:V)
(5) Solto, por si, cidadão, é que não tem diabo nenhum. (GS:V)

O desafio é, portanto, acomodar esses fenômenos naturais da constituição dos enunciados linguísticos nos aparatos metodológico-descritivos utilizados em PLN,

como é o modelo de Dependências Universais, adotado nos *treebanks* da maioria das línguas atualmente, incluindo o português brasileiro e que contempla estruturas 'canônicas' para basear e exemplificar suas diretrizes. Iniciativas recentes, todavia, vêm ampliando o leque de tipos de textos anotados para construir treebanks visando aplicações de PLN em português basileiro (DiFilippo et al., 2021; Souza et al., 2021). Nessa perspectiva, este estudo explora o potencial de um texto literário não canônico para expandir e diversificar os *treebanks* existentes, como se detalha a seguir.

3 Metodologia

O córpus anotado e analisado consiste numa amostra das primeiras 175 sentenças do romance. As sentenças foram extraídas de um arquivo em formato pdf, convertidas para o formato txt codificação UTF8 e revisadas manualmente para corrigir problemas de conversão. Para a anotação do córpus, foi utilizado o *framework* do projeto *Universal Dependencies* (UDs) v.2 (Nivre et al., 2015), que consiste em 17 etiquetas para anotação de classes gramaticais e 37 etiquetas de relações sintáticas, além de sub-relações. O córpus foi primeiramente anotado de forma automática por meio da ferramenta UDpipe (Straka et al. 2016), com um modelo de língua portuguesa que utiliza o Bosque-UD v. 2.6 de textos jornalísticos (Rademaker et al. 2017). Os arquivos CONLLU foram importados na ferramenta de anotação Arborator-Grew-NILC (https://arborator.icmc.usp), uma versão expandida e aprimorada de Arborator-Grew (Guibon et al., 2020).

A revisão da anotação automática foi realizada por 3 anotadores familiarizados com a abordagem das UDs. Para a anotação, foram utilizadas, além das diretrizes gerais das UDs, os manuais de anotação do ICMC/USP (Duran 2021a,b). Para embasar a interpretação do texto de Guimarães Rosa, foram consultadas obras sobre a linguagem e o estilo do autor (Rocha, 2021; Sant'Anna Martins, 2021). Nos casos de palavras e funções que podem operar localmente na sentença como ADV e 'advmod' ou como CCONJ e 'cc' numa relação de coordenação com uma sentença anterior, as decisões de anotação foram tomadas considerando-se o co-texto da sentença em pauta, isto é, a sentença anterior.

A revisão foi feita de forma independente pelos três anotadores, ao cabo da qual as divergências foram discutidas em grupo até se chegar a uma anotação final consensual.

Concluída a revisão, os arquivos em formato CONLLU foram exportados da ferramenta Arborator-Grew-NILC e processados por script em linguagem Python para contagem das categorias anotadas. Por meio da biblioteca Scikit-learn, foram computados a porcentagem de precisão, *recall*, *F1-score* e *support* (número total de ocorrências da etiqueta na anotação revisada) para cada categoria, juntamente com a acurácia do modelo utilizado.

A análise enfocou o percentual de acerto do modelo de anotação automática comparado com a nossa revisão manual visando verificar quais categorias foram anotadas de forma mais e menos bem sucedida e o que esses resultados poderiam evidenciar sobre o potencial do texto anotado para expandir e diversificar os treebanks em português brasileiro.

4 Resultados

A amostra de texto selecionada para anotação compreendeu 175 sentenças e 2809 *tokens*, incluindo-se sinais de pontuação. A média de *tokens* por sentença foi 16,05, tendo sido verificado um amplo intervalo de variação no tamanho mínimo e máximo das sentenças anotadas, 2 e 83 *tokens*, respectivamente.

No que diz respeito à anotação de classe de palavras, os resultados da anotação automática e sua revisão pelos anotadores estão dispostos na Tabela 1. A Tabela 1 mostra um alto índice de acerto do modelo de anotação automática, com *F1-score* maior que 85 por cento para a maior parte das categorias, com exceção de INTJ (interjeição), PART (partícula) e X (reservada para *tokens* aos quais não se pode atribuir nenhuma das categorias existentes). No caso das duas últimas, trata-se de ocorrências nas quais o modelo classificou de forma inadequada interjeições (como "eh" e "ah") e alguns sinais de pontuação. Por outro lado, a categoria INTJ teve uma alta precisão, mas baixo *recall*, o que pode ser explicado pelo fato de o modelo possuir um número limitado de lemas para a classe interjeição, os quais não contemplam as formas utilizadas por Guimarães Rosa.

Tabela 1. Taxas de precisão, *recall, F1-scor*e e *support* computadas para a anotação automática de POS.

POS	precision (%)	recall (%)	f1-score (%)	support
ADJ	85,44%	86,27%	85,85%	102
ADP	99,29%	99,29%	99,29%	283
ADV	89,30%	95,43%	92,27%	175
AUX	96,10%	100,00%	98,01%	74
CCONJ	100,00%	91,14%	95,36%	79
DET	99,36%	99,04%	99,20%	314
INTJ	100,00%	6,67%	12,50%	15
NOUN	96,32%	95,49%	95,91%	466
NUM	80,00%	100,00%	88,89%	12
PART	0,00%	0,00%	0,00%	0
PRON	98,45%	95,02%	96,71%	201
PROPN	94,92%	91,80%	93,33%	61
PUNCT	100,00%	98,98%	99,49%	587
SCONJ	93,41%	95,51%	94,44%	89
VERB	92,64%	96,87%	94,71%	351
X	0,00%	0,00%	0,00%	0

No que diz respeito às relações de dependência, a Tabela 2 sintetiza os principais resultados obtidos. Dentre as relações que tiveram um *F1-score* inferior a 75% destacam-se, para efeitos da argumentação deste estudo, as relações de *parataxe* (53,54%), *discourse* (30,77%), *orphan*, *reparandum*, *dislocated* e *dep*, estas quatro

últimas com uma única ocorrência na amostra, que o modelo não classificou de forma correta.

Tabela 2. Taxas de precisão, *recall*, *F1-score* e *support* computadas para a anotação automática de relações de dependência.

deprel	precision (%)	recall (%)	f1-score (%)	support
acl	72,73%	70,59%	71,64%	34
acl:relcl	76,32%	96,67%	85,29%	30
advcl	65,00%	61,90%	63,41%	42
advmod	78,53%	92,67%	85,02%	150
amod	81,16%	80,00%	80,58%	70
appos	48,28%	87,50%	62,22%	16
aux	91,67%	100,00%	95,65%	11
aux:pass	33,33%	100,00%	50,00%	1
case	97,83%	97,83%	97,83%	277
cc	100,00%	88,89%	94,12%	81
ccomp	53,66%	88,00%	66,67%	25
compound	0,00%	0,00%	0,00%	0
conj	83,65%	76,99%	80,18%	113
cop	83,33%	91,84%	87,38%	49
csubj	66,67%	66,67%	66,67%	6
dep	0,00%	0,00%	0,00%	1
det	99,35%	98,07%	98,71%	311
discourse	66,67%	20,00%	30,77%	30
dislocated	0,00%	0,00%	0,00%	2
expl	93,33%	56,00%	70,00%	25
fixed	57,14%	40,00%	47,06%	20
flat:name	85,71%	100,00%	92,31%	6
iobj	92,31%	70,59%	80,00%	17
mark	85,06%	94,87%	89,70%	78
nmod	86,61%	83,62%	85,09%	116
nsubj	84,83%	90,96%	87,79%	166
nsubj:pass	0,00%	0,00%	0,00%	1
nummod	91,67%	100,00%	95,65%	11

obj	74,42%	85,71%	79,67%	112
obl	86,73%	74,81%	80,33%	131
orphan	0,00%	0,00%	0,00%	1
parataxis	64,15%	45,95%	53,54%	74
punct	100,00%	98,98%	99,49%	587
reparandum	0,00%	0,00%	0,00%	1
root	86,86%	86,86%	86,86%	175
xcomp	68,09%	82,05%	74,42%	39

A Tabela 3 apresenta os valores médios obtidos para cada conjunto de etiquetas. Nela observamos performance superior do modelo de máquina para POS com acurácia de 92,26% em contraste com 88,22% para relações de dependências. No entanto, ao avaliar o F1-score obtido, a taxa de acerto cai em média 25 por cento para ambas, demonstrando uma performance mediana especialmente para as relações de dependência. Quanto às medidas de precisão e recall, percebe-se uma leve diferença na avaliação das POS, que pode ser justificada pela dificuldade do modelo em reconhecer os tokens que deveriam receber a etiqueta INTJ e pelo reconhecimento errôneo das etiquetas PART e X, como mencionado previamente.

Tabela 3. Média das taxas de precisão, *recall*, *F1-score* e acurácia calculadas para a anotação automática de cada conjunto de etiquetas.

	precisão (%)	**recall (%)**	**F1-score (%)**	**acurácia (%)**
POS	82,83%	78,22%	77,87%	96,26%
deprels	65,42%	67,44%	64,95%	88,22%

As relações que tiveram um *F1-score* inferior a 75% revelam mecanismos típicos da oralidade, como disfluências, interposição e deslocamento de constituintes, bem como relações cuja dependência não pode ser classificada de acordo com nenhuma categoria já estabelecida pelo modelo das UDs. Os exemplos a seguir ilustram algumas dessas formas e os desafios que apresentam para sua anotação.

A Figura 1 mostra um exemplo no qual se verifica falta de alinhamento entre uma classe de palavras e a relação de dependência da qual participa. O advérbio “depois” é utilizado na sentença, não numa relação *advmod,* mas numa relação *obj*, sendo *head e* possuindo um determinante.

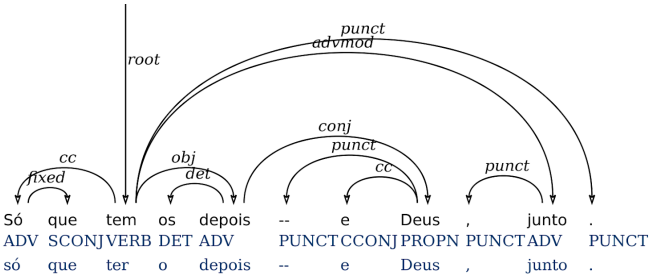

Figura 1. Exemplo de incongruência entre classe e função.

A Figura 2 mostra uma sentença com uma relação *ccomp* envolvendo uma relação na qual um deslocamento de parte de um sintagma nominal demanda o estabelecimento de uma relação não projetiva.

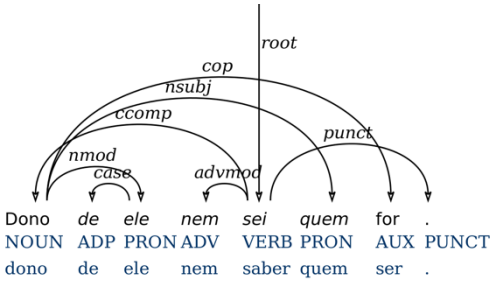

Figura 2. Exemplo de uma anotação com relação não projetiva.

A Figura 3 mostra a anotação de um fragmento de uma sentença, na qual se verifica uma relação de dependência não especificada (*dep*). De acordo com Rocha (2021), "elas se acostumaram a se assim das locas, para papar" envolve o apagamento do verbo "sair". Essa ausência impede atribuir a "se" uma relação específica.

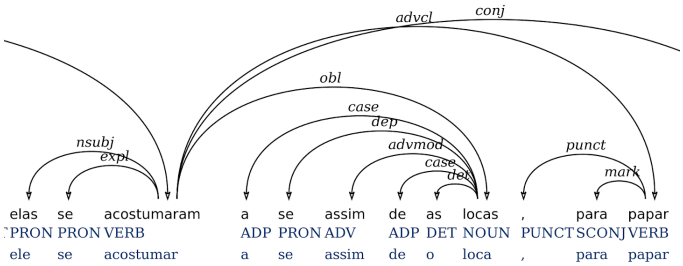

Figura 3. Exemplo da relação *dep*.

5 Discussão

A acurácia do modelo de anotação automática da amostra em questão evidencia, por um lado, o bom desempenho de modelos baseados em textos canônicos, como é o texto jornalístico, para a anotação de textos menos canônicos; por outro, mostra o potencial

do texto não canônico para complementar os modelos existentes em categorias menos representadas nos *treebanks*, evidenciado pelo número pequeno de lemas existentes para INTJ e de instâncias para as relações *discourse, reparandum e dislocated* conforme dados disponibilizados em https://universaldependencies.org/treebanks/pt_bosque/index.html.

Os desafios de anotação, no sentido de total aderência às diretrizes do projeto UD, concentram-se em (i) o não alinhamento entre classe de palavra e função e (ii) configurações sintáticas que demandam anotação não projetiva. Considerando-se que os treebanks disponíveis em português brasileiro que embasam o treinamento do modelo utilizado neste estudo e de outros modelos, em geral, ainda se baseiam em textos canônicos, é possível dizer que os recursos para PLN nesse idioma podem beneficiar-se da anotação de textos não canônicos que contemplem tais desafios, de modo a incrementar a construção de recursos em português brasileiro para PLN.

6 Considerações finais

Este estudo exploratório de *Grande sertão: Veredas* procurou mostrar a pertinência de se incluir textos não-canônicos, dentre eles literários, aos *treebanks* para PLN em português brasileiro. A justificativa para a escolha da amostra é sua representação do registro oral em português, fornecendo dados para expandir o repertório de anotação em português brasileiro pautado pelas Dependências Universais. Longe de representar um desvio, a linguagem rosiana, não canônica do ponto de vista do PLN, tem como matriz o sistema do português brasileiro e muitas das estruturas e do léxico nela presentes permitem incorporar fenômenos da oralidade aos *treebanks*.

Esses fenômenos, tão naturais da língua falada, e, no córpus desta pesquisa, registrado na modalidade escrita, são pervasivos em inúmeros gêneros discursivos em que se verifica essa relativização de fronteiras entre o oral e o escrito (Marcuschi, 2008, 2010; Neves, 2010), como crônicas, postagens em redes sociais e em *blogs*, tipos de textos alvejados em projetos de anotação em PLN.

O corpus de sentenças anotadas está em processo de preparação para sua validação e submissão à comunidade UDs e será também disponibilizado na conta de *github* dos pesquisadores.

Referências

Abaurre, M. B. M., Rodrigues, A. C. S. (orgs.): Gramática do português falado. Vol. 8 – Novos estudos descritivos. Editora Unicamp, Campinas (2002).
Bamman, D.: LitBank: Born-Literary natural language processing.
Castilho, A. T. (org.): Gramática do português falado. Vol. 1 – A ordem. 4a ed. Editora Unicamp, Campinas (2002a).
Castilho, A. T. (org.): Gramática do português falado. Vol. 3 – As abordagens. 3a ed. Editora Unicamp, Campinas (2002b).
Castilho, A. T., Basílio, M. (orgs.): Gramática do português falado. Vol. 4 – Estudos descritivos. 2a ed. Editora Unicamp, Campinas (2002a).
Di Felippo, A. et al.: Descrição Preliminar do Corpus DANTEStocks: Diretrizes de Segmentação para Anotação segundo Universal Dependencies. In: the Proceedings of the VII Workshop on Portuguese Description (JDP), pp. 335-343. (2021).

Duran, M. S.: Manual de anotação de PoS tags. Relatório Técnico, n. 434. NILC-ICMC/USP, 54p. (2021a) Disponível em: https://sites.google.com/icmc.usp.br/poetisa. Acesso em: 20/09/2021.
Duran, M. S.: Manual de Anotação de Relações de Dependência: Orientações para anotação de relações de dependência sintática em Língua Portuguesa, seguindo as diretrizes da abordagem Universal Dependencies (UD). Relatório Técnico do ICMC 435. Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo. São Carlos-SP, 79 p. (2021b).
Heine, B., Kaltenböck, G., Kuteva, T., Long, H.: An outline of Discourse Grammar. In: Bischoff, S., Jeny, C. (eds.) Reflections on functionalism in linguistics, pp. 141-157, Mouton de Gruyter, Berlin (2013).
Heine, B., Kaltenböck, G., Kuteva, T., Long, H.: Cooptation as a discourse strategy. Linguistics 55(4), 813-855 (2017)
Hirschmann, H., Doolittle, S., Lüdeling, A.: Syntactic annotation of non-canonical linguistic structure. (2007)
Ilari, R. (org.): Gramática do português falado. Vol. 2 – Níveis de análise. 4a ed. Editora Unicamp, Campinas (2002).
Kaltenböck, G., Heine, B., Kuteva, T.: On thetical grammar. Studies in Language 35(4), 852-897 (2011).
Kato, M. A (org.): Gramática do português falado. Vol. 5 – Convergências. 2a ed. Editora Unicamp, Campinas (2002).
Koch, I. (org.): Gramática do português falado. Vol. 6 – Desenvolvimentos. 2a ed. Editora Unicamp, Campinas (2002).
Marcuschi, L. A.: Produção textual, análise de gêneros e compreensão. Parábola Editorial, São Paulo (2008).
Marcuschi, L. A.: Da fala para a escrita. Atividades de retextualização. 10a ed. Cortez Editora, Campinas (2010).
Martins, N. S.: O léxico de Guimarães Rosa. 3a ed. Edusp, São Paulo (2001).
Marneffe, M. C., Manning, C. D., Nivre, J., Zeman, D. Universal dependencies. Computational Linguistics 47(2), 255-308 (2021).
Neves, M. H. M.: Língua falada e língua escrita. Uma busca da gramática que rege as formulações. In: Neves, M. H. M. Ensino de língua e vivência de linguagem: temas em confronto, p. 151-167, Editora Contexto, São Paulo (2010).
Neves, M. H. M.: As estratégias discursivas e suas implicações na relação entre oralidade e escrita – um estudo do parêntese na crônica. Linguística 27(1), 77-97 (2012).
Neves, M. H. M. (org.): Gramática do português falado. Vol. 7 – Novos estudos. 2a ed. Editora Unicamp, Campinas (1999).
Nivre, J. et al.: Universal Dependencies v2: An ever growing multilingual treebank collection. In: Proceedings of the12th Language Resources and Evaluation Conference, 4034-4043. Marseille, France: European Language Resources Association. (2020)
Plank, B.: What to do about non-standard (or non-canonical) language in NLP. In: Proceedings of the 13th Conference on Natural Language Processing (KOVENS2016), p. 13-20. NLP Association of India, Varanasi (2016).
Rademaker, A. et al.: Universal dependencies for portuguese. In Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017), pages 197–206. (2017).
Rosa, J. G.: Grande sertão: Veredas. 22a ed. São Paulo: Companhia das Letras (2019/1959).
Silveira, N. et al.: A gold standard dependency corpus for English. In: Proceedings of the 9th International Conference on Language Resources and Evaluation, p. 2897-2904. European Language Resources Association, Reykjavik (2014).
Souza, E. et al.: PetroGold – Corpus padrão ouro para o domínio do petróleo. *In*: Anais do 13º Simpósio Braisleiro de Tecnologia da Informação e da Linguagem Humana (STIL), p. 29-38, Porto Alegre: Sociedade Brasileira de Computação (2021).

Straka, M., Hajic, J., Strakov´a, J.: Udpipe: trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16), 4290–4297 (2016).
Tarallo, F., Kato, M. et al.: Rupturas na ordem da adjacência canônicas no português falado. In: Castilho, A. (org.) Gramática do português falado. Vol. 1 – A ordem. 1a ed., pp. 25-52, Editora Unicamp, Campinas (1989).
Zeldes, A., Simonson, D.: Different flavors of GUM: Evaluating genre and sentence type effects on multilayer corpus annotation quality. In: Proceedings of the 10th Linguistic Annotation Workshop, p. 68-78. Association for Computational Linguistics, Berlin (2016).

Polishing the gold – how much revision do we need in treebanks? ⋆

Elvis de Souza and Cláudia Freitas

Pontifical Catholic University of Rio de Janeiro
elvis.desouza99@gmail.com,
claudiafreitas@puc-rio.br

Abstract. We present the second version of PetroGold, a gold-standard treebank for the oil & gas domain in the Portuguese language. The corpus went through a series of revisions guided by three methods tested in the literature: inter-annotator disagreement, inconsistent n-grams and verification rules. We perform an intrinsic evaluation and the model scores 90.92%, 89.09% and 84.07% in the UAS (unlabeled attachment score), LAS (labeled attachment score) and CLAS (content-word labeled attachment score) metrics respectively, CLAS being 1.11% higher than in the first version. We perform an experiment where we verify a negative impact in the intrinsic evaluation when simplifying the annotation related to prepositional verbal arguments and we conclude by discussing the results and future work.

Keywords: Natural Language Processing · Language resources · Corpora reviewing · Treebank

1 Introduction

Annotated corpora are important resources for natural language processing. On the one hand, data-driven NLP approaches use corpora as a learning source for linguistic analysis; on the other hand, approaches based on rules, or oriented by specific knowledge of language, can use it as material to evaluate the results of their analyses. Despite its importance, the number of golden treebanks in Portuguese, with texts from genres other than the journalistic one, still falls short, making it difficult to advance certain NLP tasks for Portuguese, such as information extraction and parsing in diverse domain areas.

⋆ This paper was partially funded by the National Agency for Petroleum, Natural Gas and Biofuels (ANP), Brazil, associated with the investment of resources from the R, D & I Clauses, through a Cooperation Agreement between Petrobras and PUC-Rio. We would like to thank the team at the Applied Computational Intelligence Laboratory (ICA) at PUC-Rio for the generation of morphosyntactic annotation models trained in Stanza, and Elvis de Souza thanks the National Council for Scientific and Technological Development (CNPq) for the Masters scholarship process no. 130495/2021-2.

Some of the main machine-learning-based parsers available to the Portuguese-speaking community (e.g. spaCy [10], UDPipe [24] and Stanza [17]) use the Bosque-UD [18] corpus as training material, achieving a performance of up to 87.81% in the attachment of dependencies, according to the CoNLL 2018 Shared Task [28]. The corpus is composed of morphosyntactically annotated journalistic texts and is part of the Universal Dependencies [14] project, representing a valuable resource for several NLP tasks in general domain texts.

However, tasks that demand the processing of texts from specific domains will face difficulties due to the lack of available training material of diverse domains – fortunately, this scenario is changing with the creation of projects like Porttinari [15]. [25] indicates that, for the English language, a model trained in the Wall Street Journal Treebank sees its performance drop more than 10% when applied in the biomedical domain. Similarly, [6] reports that systems trained with general domain texts do not perform well when applied to academic texts.

In this paper, we present a second version of PetroGold, a gold-standard treebank with texts from the oil & gas domain in the academic genre. The corpus is available at the project webpage[1] and contains 8,949 sentences (250,595 tokens). More than providing an improved version of the material, which includes a systematic treatment of tags related to verbal subcategorization and grammatical multiword expressions, this second round of review aims to (i) evaluate the contribution of different treebank review methods in a robust corpus, offering subsidies for an evaluation of the methodology described in [8], and (ii) evaluate how much the corrections carried out in this second stage impact the performance of language models. In the end, we assess how much differentiating adverbial adjuncts from prepositional verbal arguments impacts the performance of a parser.

2 Treebanks

Syntactically analyzed corpora are called treebanks because syntactic analyses give a hierarchical character to sentences using constituents (in a syntagmatic model) or dependents (in a dependency model). From the point of view of linguistic studies, treebanks are informative about the structure of the language in use, serving as a database for the development of linguistic theories, either as a means of testing them or in carrying out statistical studies. From a NLP point of view, treebanks serve as training and evaluation material, in addition to being the basis for subsequent tasks, such as Open Information Extraction [9].

There are many possible differences between treebanks, some related to the methodology for building the corpus – annotated entirely from scratch or, more commonly in recent times, automatically annotated and revised by linguists – and differences related to the syntactic categories, to the grammar model and others. The pioneering English-language corpora, Penn Treebank [12] and SUSANNE [19], made their syntactic annotation available through constituents;

[1] https://petroles.puc-rio.ai

the Prague Dependency Treebank [4], in turn, uses a syntactic dependencies format.

For the Portuguese language, Linguateca [20] has been dedicated for a long time to the creation of Floresta Sintá(c)tica ([1], [7]), a pioneering project for the construction of treebanks for the Portuguese language. Floresta obtained its morphosyntactic annotation from the automatic analyzer PALAVRAS [3] and it is composed of four parts, which differ in terms of modality (written or spoken) and degree of revision. Bosque is a subset of Floresta, is fully revised by linguists, and it is precisely the revision dimension that made (and makes) Bosque a valuable resource, which is reflected in its conversion to different formats, such as Bosque-UD [18].

3 Building PetroGold v2

PetroGold is composed of 8,949 sentences (250,595 tokens) from 19 theses and dissertations of the oil & gas domain processed in full: only elements such as summary, abstract, appendices and bibliographic references were excluded, as well as figures, graphs, formulas and tables. The corpus was annotated using the Universal Dependencies framework. However, since issues related to the academic genre and the specific technical domain are not covered in the project's annotation guidelines, we needed to discuss how to carry out the analysis of the typical linguistic structures of the corpus. Some of the new problems that have arisen since the release of the first version of the corpus, in addition to the methodology and tools used in the development of this new version will be discussed in this section.

3.1 Annotation challenges

The second version of PetroGold brings at least three major improvements related to the annotation of grammatical multiword expressions, verbal lemmatization and verbal subcategorization.

First, we standardized the annotation of grammatical multiword expressions (MWEs) such as *de acordo com* ("according to"), *por sua vez* ("in turn") and *tendo em vista* ("in view of"), which receive the *fixed* dependency relation. We used as a criterion the recognition of combinations as grammatical phrases (prepositional, conjunctive, adverbial) in Portuguese language grammars and the difficulty of dealing with the combination in a transparent manner, both at the part-of-speech and in the syntactic level. A complete listing of these 227 expressions, which as a whole occur 2,333 times in the corpus, can be found in the documentation accompanying the corpus.

Another improvement is related to the lemmatization of verbs. Since PetroGold was originally annotated by a system trained using a journalistic corpus, many of the verbs specific to the academic genre and the oil & gas domain were not correctly identified by the model, such as *adsorver* ("to adsorb"), lemmatized as *adsorvir* and *absorver* ("to absorb"), lemmatized as *absorvar* – both

very common in the technical domain. In this second phase, we performed a manual verification of all verbal lemmas in the corpus, resulting in 212 corrected lemmas, which occur 621 times in the corpus.

This version also features many corrections regarding adjuncts and verbal arguments, a topic which is thoroughly discussed in [22]. The grammatical guidelines of the Universal Dependencies project for this issue follow the direction of [27]: given the difficulties of distinguishing argument and adjunct already known and reported in the literature ([11], [16], [26], [2]) – and that difficulties are common in corpus annotation at least for most of the languages that make up the project – the project chooses to (partially) shift the discussion to another place: the idea is not to distinguish argument from adjunct, but between the core and the oblique terms.

In short, when related to verbal subcategorization, the core terms are not introduced by preposition and the tags are *obj* and *iobj* – the latter only used with arguments that are oblique pronouns – and oblique terms are preceded by a preposition and the tag is *obl*). However, UD also allows us to annotate a subspecification of the oblique, *obl:arg*, when, in addition to being prepositional, the phrase is also an argument of the verb, if we find the distinction to be important.

While analyzing *obl* and *obl:arg* in PetroGold, we do not seek to characterize arguments based on the transitivity of the verb, but we prioritize the meaning of the prepositional phrase – if it expresses meanings traditionally associated with adverbials (time, place, manner, purpose, causality, conformity etc), we annotate as *obl*, while, in the absence of an adverbial semantics, we analyze as *obl:arg*.

3.2 Methodology

The first version of PetroGold had four annotators working 20 hours a week, for three months, dedicated to reviewing the corpus. In this second version, we had three of the annotators working 20 hours a week for two months. All annotators had previously familiarized themselves with both the UD approach and the type of text that makes up the corpus.

The corpus was originally annotated using a customized Stanza model, which was trained using Bosque-UD plus a small portion of sentences from other texts of the domain with totally revised annotation[2]. The inter-annotator agreement in the human review of the automatic annotation was 95.1% using the κ (*kappa*) metric for the pair of annotators that obtained the highest degree of agreement in the syntactic dependency analysis task, while the worst performing pair obtained 91.9% agreement.

The first version of the corpus used as a review strategy the analysis of confusion matrices, which contrast the analysis of two different parsers, Stanza and UDPipe, in such a way that the divergences between both systems are indicative of possible errors in one of the systems or both, requiring human intervention to choose the correct analysis[3]. This strategy, which we call IAD (Inter-Annotator

[2] In this training material, Bosque-UD represented 93% of the total size.

[3] Since both analysis systems can perform tokenization and sentence segmentation in different ways, we gave UDPipe the corpus already segmented by Stanza.

Disagreement), allows the analysis of errors by clusters of confusion between parsers, making it easier for annotators to detect error patterns and, consequently, to develop different correction rules.

For the second version of PetroGold, we applied the revision strategy schematized in [8], which consists, in addition to the IAD method, in the verification of inconsistent n-grams and the application of general verification rules.

Inconsistent n-grams is a method proposed by [5] and adapted to UD by [13]. The underlying idea is that a pair of dependent lemmas, if repeated in the corpus, must have the same annotation in all occurrences, otherwise it is indicative of annotation inconsistency. For example, in sentences (3) and (4), the same pair of lemmas (**Arai** and **1990**) was analyzed differently in two sentences: in the first, the analysis is of a composite proper name (*flat:name*), and in the second, the analysis is of an adnominal adjunct. Bibliographic references that contain the publication year have the relation between the date and the proper name analyzed as *nmod*, so sentence (3) needed to be corrected to become consistent with the analysis of (4), which is the correct one.

(3) ***flat:name*** – Da mesma forma, **Arai** & Coimbra (**1990**) interpretam que o paleoambiente do Membro Romualdo (...)[4]

(4) ***nmod*** – Desta forma, a característica geral da associação fossilífera encontrada por **Arai** & Coimbra (**1990**) não deixa dúvidas quanto à pertinência dos registros das ingressões marinhas no Andar Alagoas.[5]

Differently from previous authors, we did not require that the words in the context of the pairs should be the same in order to look for divergent pairs annotation because it lowered the method recall.

The other method, a rule-based verification approach, consists of search expressions created to detect errors in the corpus, whether referring to inconsistencies regarding the UD format or the Portuguese annotation.

For example, the comma in (5) after the expression *a seguir* ("next") depended on the verb *seguir* ("to follow"); however, since it is a multiword expression (*fixed*), the comma should depend on the head of the expression, "a", a restriction of the UD model. In (6), the occurrence of a verb in the participle, *denominada* ("denominated"), with a verb *ser* ("to be") depending on it, is typical of passive voice; thus, *ele* ("he"), which is introduced by the preposition *por* ("by"), is a common form of agent of the passive voice in Portuguese, although it had not been analyzed in this way.

(5) **A seguir,** são apresentadas as etapas e a metodologia que foi adotada no trabalho.[6]

[4] Transl. "Similarly, **Arai** & Coimbra (**1990**) interpret that the paleoenvironment of the Romualdo Member (...)"

[5] Transl. "In this way, the general characteristic of the fossiliferous association found by **Arai** & Coimbra (**1990**) leaves no doubt as to the relevance of the records of marine ingressions in the Andar Alagoas."

[6] Transl. "**Next,** the steps and methodology adopted in the work are presented."

(6) Esta zona de falha foi por **ele** denominada "Zona de Transferência de o Funil".[7]

A list with all 61 rules can be found on our GitHub page[8].

In order to evaluate the impact of the revisions, in section 4 we compare three different learning scenarios: (a) the first version against this second revised version, and (b) the second version, which has the *obl:arg* annotation, against the same corpus when the tag is converted to *obl*, simulating what the UD guidelines first suggest.

3.3 Tools

The review was performed using ET, a tool that enables querying, editing and evaluating annotated corpora [21]. ET is divided into Interrogatório, an interface where we search for the most frequent errors and correct their annotation using the correction rules that we developed during the review process, and Julgamento, an interface where we evaluate the linguistic annotation according to the aforementioned methods: inter-annotator disagreement, inconsistent n-grams and verification rules.

Besides seeing the annotation from both a quantitative and a qualitative perspective (for instance, the main tags involved in annotation errors), reading the corpus through the lens of Julgamento provides us with a picture of strengths and weaknesses of the annotation. This picture, in turn, guides us back to Interrogatório: we can search for the same sentences pointed out in the review methods and make corrections manually or in batch, when applicable, to make the review more efficient.

4 Results

PetroGold v2 is slightly smaller than the first version due to some sentences that were suppressed because of incorrect segmentation in the pre-processing stage. Table 1 indicates the differences in the characteristics of both versions of the corpus.

In this second version, the number of tokens corrected since the original annotation from Stanza (summing versions 1 and 2) reached 21,634 – 8.6% of all tokens needed correction –, resulting in 74% of sentences which had at least one token modified by the annotators.

Regarding the review methods described in Section 3.2, Figure 1 illustrates the contribution of each of them in the review process. The figure is an estimation of the relative number of errors found by each method because, since two or more methods can indicate the same token as an annotation mistake, the number of errors found by all methods, when summed, exceed 100% of the corrected tokens.

[7] Transl. "This fault zone was called by **him** 'the Funnel Transfer Zone'".

[8] Available at: https://github.com/alvelvis/ACDC-UD/blob/master/validar_UD.txt. Accessed on 15 Jan. 2022.

	v2	v1
Tokens	250,595	253,640
Corrections	8,802	12,832
Words	221,208	223,707
Sentences	8,949	9,127
Documents	19	19

Table 1. PetroGold features across versions

The most productive method for identifying errors was IAD, which sums up 51.4% of detected errors (11,137 tokens). The general correction rules, in turn, totaled 10.1% (2,202 tokens), while the inconsistent n-grams indicated 9.2% of the corrected errors (2,003 tokens). None of the methods was able to identify 37.8% of the errors (8,188 of the tokens), which were found by the annotators when reading the sentences in the treebank.

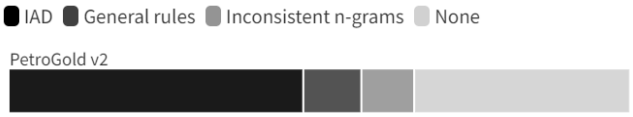

Fig. 1. Methods contribution

From the reviewing process point of view, the IAD method spots the highest number of annotation mistakes. Since we previously showed [8] that this approach also achieves the best F1 among the revision methods (49.7%, 14.4% and 4.8% for IAD, rule-based and inconsistent n-grams, respectively), choosing only IAD is a possibility to be considered when there is not much time or resources to build a revised treebank.

To check the material's consistency, we trained a UDPipe model using the tool's default parameters and PetroGold v2 as the training material. We used the same set of sentences from the experiment performed in [23] in the train and test partitions to allow comparisons between the results from both versions 1 and 2 of the treebank, following the proportion of 95% and 5% of sentences in each partition, respectively. Previously, the performance of the model was compared against the results of a model trained using Bosque-UD, which has a similar size – at that time, PetroGold v1 achieved up to 9% better metrics than the journalistic corpus [23]. This time, we compared the second version of PetroGold, with all the corrections reported, with the first version of the corpus.

As seen in Table 2, the results for PetroGold v2 are not very different from those of the first version with regard to the lemmatization task (LEMMA), part-of-speech assignment (UPOS), syntactic dependency attachment (UAS), classification of dependencies (LAS) and classification of dependencies for content

words (CLAS)[9]. We see a slight improvement in all metrics: LEMMA (0.06%), UPOS (0.21%), UAS (0.27%), LAS (0.56%) and CLAS (1.11%).

Version	LEMMA (%)	UPOS (%)	UAS (%)	LAS (%)	CLAS (%)
v2	98.54	98.40	90.92	89.09	84.07
v1	98.48	98.19	90.65	88.53	82.96

Table 2. Intrinsic evaluation of PetroGold models trained using UDPipe

The results show a modest increase in consistency, suggesting that intrinsic evaluation will not be very sensitive to corpora reviews when they were already internally consistent. The main increase is on CLAS measure, and this can be due to the large review applied to prepositional verbal arguments and adjuncts, which are content words.

Finally, we compared the PetroGold v2 intrinsic evaluation results with the results from the same corpus having converted the *obl:arg* tags to *obl*. In this version, which we call "No *obl:arg*", there is no distinction between adverbial adjuncts and prepositional objects, so that every verb-dependent prepositional phrase receives the tag *obl*, as originally proposed in [27]. It is a modification that affected 1,488 tokens, which are present in 14.8% of sentences.

	LEMMA (%)	UPOS (%)	UAS (%)	LAS (%)	CLAS (%)
No *obl:arg*	98.54	98.40	90.66	88.82	83.48

Table 3. Model evaluation when *obl:arg* is converted to *obl*

The results in Table 3 indicate a drop in all metrics related to dependency analysis (UAS, LAS and CLAS), with emphasis on CLAS, whose performance drop was 0.59%. At first sight, the results seem counter-intuitive, as the *obl:arg* tag represents a semantically oriented analysis, thus a more difficult one – the same phrase introduced by preposition can receive either one tag or another, depending on the meaning of the phrase content words, while the simplified version would be analyzed one way or another based only on the presence of a preposition. However, maintaining this granularity – the distinction between argument and adjunct in prepositional phrases – facilitated the generalization of the system as a whole, indicated by the decrease in all metrics when the distinction is undone.

The results by each dependency relations show that the *obl:arg* is a difficult one (only 62.8% hits) and that the *obl* relation is best learned when we convert all the *obl:arg* relations to *obl* (86.4% against 79.9%). However, many other

[9] Metrics were gathered from [28].

dependency relations were best learned when we had the *obl:arg* relation in the treebank, such as *ccomp* (65.5% with *obl:arg* against 58.6% without it) and *acl:relcl* (95.6% against 93.4%). While there is no direct explanation for the dependency relations hit increase with *obl:arg*, it justifies the overall better F1, which adds up to the many arguments in favor of keeping this annotation in the corpus.

5 Concluding remarks

We presented a small study on treebank revision methods, based on PetroGold corpus. While the most productive review method spots around half of the annotation mistakes, almost 40% of them can not be detected by any method. Besides presenting PetroGold v2, we performed an intrinsic evaluation of annotation consistency, and compared PetroGold v2 against PetroGold v1, which resulted in a 1.11% increase in CLAS. We concluded by confirming that an intrinsic evaluation might not be sensitive to improvements in robust corpora that have previously been reviewed, in spite of the importance of improving some specific annotations for different reasons. Furthermore, we verified that removing the distinction between adverbial adjuncts and prepositional verbal arguments has a slight negative impact in the automatic learning of dependencies.

The first application of PetroGold will be the creation of a morphosyntactic annotation model suitable for texts in the oil & gas domain. The goal is to use this customized model to annotate new texts in the domain in order to proceed with a semantic annotation of named entities, increasingly expanding the coverage of Portuguese NLP directed to specific domains.

References

1. Afonso, S., Bick, E., Haber, R., Santos, D.: Floresta sintá(c)tica: a treebank for Portuguese. In: Rodrigues, M.G., Araujo, C.P.S. (eds.) Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002). pp. 1698–1703. ELRA, Paris (29-31 de Maio 2002), http://www.linguateca.pt/documentos/AfonsoetalLREC2002.pdf
2. Bagno, M.: Gramática pedagógica do português brasileiro. Parábola Ed. (2012)
3. Bick, E.: Palavras, a constraint grammar based parsing system for portuguese. Working with Portuguese corpora pp. 279–302 (2014)
4. Böhmová, A., Hajič, J., Hajičová, E., Hladká, B.: The prague dependency treebank. In: Treebanks, pp. 103–127. Springer (2003)
5. Boyd, A., Dickinson, M., Meurers, W.D.: On detecting errors in dependency treebanks. Research on Language and Computation **6**(2), 113–137 (2008)
6. Cohen, K.B., Verspoor, K., Fort, K., Funk, C., Bada, M., Palmer, M., Hunter, L.E.: The colorado richly annotated full text (craft) corpus: Multi-model annotation in the biomedical domain. In: Handbook of Linguistic Annotation, pp. 1379–1394. Springer (2017)
7. Freitas, C., Rocha, P., Bick, E.: Floresta Sintá(c)tica: Bigger, Thicker and Easier. In: Teixeira, A., de Lima, V.L.S., de Oliveira, L.C.,

Quaresma, P. (eds.) Computational Processing of the Portuguese Language, 8th International Conference, Proceedings (PROPOR 2008). vol. Vol. 5190, pp. 216–219. Springer Verlag (8-10 de Setembro 2008), http://www.linguateca.pt/documentos/FreitasRochaBickPROPOR08Poster.pdf
8. Freitas, C., de Souza, E.: A study on methods for revising dependency treebanks: In search of gold (2022), submitted
9. Gamallo, P., Garcia, M., Fernández-Lanza, S.: Dependency-based open information extraction. In: Proceedings of the joint workshop on unsupervised and semi-supervised learning in NLP. pp. 10–18 (2012)
10. Honnibal, M., Johnson, M.: An improved non-monotonic transition system for dependency parsing. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 1373–1378. Association for Computational Linguistics, Lisbon, Portugal (September 2015), https://aclweb.org/anthology/D/D15/D15-1162
11. Manning, C.D.: Probabilistic syntax. Probabilistic linguistics **289341** (2003)
12. Marcus, M., Santorini, B., Marcinkiewicz, M.A.: Building a large annotated corpus of english: The penn treebank (1993)
13. de Marneffe, M.C., Grioni, M., Kanerva, J., Ginter, F.: Assessing the annotation consistency of the Universal Dependencies corpora. In: Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017). pp. 108–115. Linköping University Electronic Press, Pisa,Italy (Sep 2017), https://www.aclweb.org/anthology/W17-6514
14. de Marneffe, M.C., Manning, C.D., Nivre, J., Zeman, D.: Universal dependencies. Computational linguistics **47**(2), 255–308 (2021)
15. Pardo, T., Duran, M., Lopes, L., Felippo, A., Roman, N., Nunes, M.: Porttinari - a large multi-genre treebank for brazilian portuguese. In: Anais do XIII Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana. pp. 1–10. SBC, Porto Alegre, RS, Brasil (2021). https://doi.org/10.5753/stil.2021.17778, https://sol.sbc.org.br/index.php/stil/article/view/17778
16. Przepiórkowski, A., Patejuk, A.: Arguments and adjuncts in universal dependencies. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 3837–3852 (2018)
17. Qi, P., Zhang, Y., Zhang, Y., Bolton, J., Manning, C.D.: Stanza: A Python natural language processing toolkit for many human languages. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (2020), https://nlp.stanford.edu/pubs/qi2020stanza.pdf
18. Rademaker, A., Chalub, F., Real, L., Freitas, C., Bick, E., de Paiva, V.: Universal dependencies for portuguese. In: Proceedings of the Fourth International Conference on Dependency Linguistics (Depling 2017). pp. 197–206 (2017)
19. Sampson, G.: English for the computer: The susanne corpus and analytic scheme (2002)
20. Santos, D., Simões, A., Frankenberg-Garcia, A., Pinto, A., Barreiro, A., Maia, B., Mota, C., Oliveira, D., Bick, E., Ranchhod, E., et al.: Linguateca: um centro de recursos distribuído para o processamento computacional da língua portuguesa (2004)
21. de Souza, E., Freitas, C.: ET: A workstation for querying, editing and evaluating annotated corpora. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 35–41. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (Nov 2021). https://doi.org/10.18653/v1/2021.emnlp-demo.5, https://aclanthology.org/2021.emnlp-demo.5

22. de Souza, E., Freitas, C.: Still on arguments and adjuncts: the status of the indirect object and the adverbial adjunct relations in universal dependencies for portuguese. In: Proceedings of the I Universal Dependencies Brazilian Festival (UDFest-BR) (2022)
23. Souza, E., Silveira, A., Cavalcanti, T., Castro, M., Freitas, C.: Petrogold – corpus padrão ouro para o domínio do petróleo. In: Anais do XIII Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana. pp. 29–38. SBC, Porto Alegre, RS, Brasil (2021). https://doi.org/10.5753/stil.2021.17781, https://sol.sbc.org.br/index.php/stil/article/view/17781
24. Straka, M., Hajic, J., Straková, J.: Udpipe: trainable pipeline for processing conll-u files performing tokenization, morphological analysis, pos tagging and parsing. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). pp. 4290–4297 (2016)
25. Thompson, P., Ananiadou, S., Tsujii, J.: The genia corpus: Annotation levels and applications. In: Handbook of Linguistic Annotation, pp. 1395–1432. Springer (2017)
26. Vilela, Mário; Koch, I.V.: Gramática da língua portuguesa: Gramática da Palavra, Gramática da Frase, Gramática do Texto/Discurso. Almedina (2001)
27. Zeman, D.: Core arguments in universal dependencies. In: Proceedings of the fourth international conference on dependency linguistics (DepLing 2017). pp. 287–296 (2017)
28. Zeman, D., Hajic, J., Popel, M., Potthast, M., Straka, M., Ginter, F., Nivre, J., Petrov, S.: Conll 2018 shared task: Multilingual parsing from raw text to universal dependencies. In: Proceedings of the CoNLL 2018 Shared Task: Multilingual parsing from raw text to universal dependencies. pp. 1–21 (2018)

Que simples que nada: a anotação da palavra *que* em córpus de UD

Magali Sanches Duran[1] Heloísa de Oliveira[2] Clarissa Scandarolli[2]

[1] Núcleo Interinstitucional de Linguística Computacional (NILC)
Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo (USP)
[2] Universidade Federal de São Carlos (UFSCAR)
{magali.duran@uol.com.br}{heloisa.oliveira@estudante.ufscar.br}
{claleni@gmail.com}

Abstract. This paper discusses the various classifications of the Portuguese language functional word *que* in order to support decisions on how to annotate it using the Universal Dependencies framework. The most frequent uses of *que* have a fairly consistent classification in grammars and dictionaries. The less frequent uses, however, are often not mentioned by many authors, and when they are, they have different classifications. The result of this research is a series of decisions made on a set of sentences illustrating various uses of *que*. The annotation of this set of sentences is available for consultation online.

Keywords: Universal Dependencies, lexical ambiguity, Portuguese language.

1 Introdução

A anotação de córpus para fins de PLN apresenta vários desafios. O mais importante deles, contudo, é garantir que fenômenos diferentes sejam anotados de formas diferentes e fenômenos iguais sejam anotados com as mesmas etiquetas. A isso se chama consistência de anotação, requisito essencial para que o aprendizado automático não seja prejudicado pela falta de qualidade na anotação do córpus de treinamento.

Para garantir uma anotação consistente, é necessário que os anotadores sejam treinados para reconhecer padrões de atribuição de etiquetas e tenham acesso a amplo material contendo diretrizes e exemplos de anotação. Além de dominar os conjuntos de etiquetas, o anotador necessita também saber lidar com a ambiguidade lexical.

Durante a anotação, é relativamente mais simples resolver a ambiguidade de palavras de conteúdo (substantivos, adjetivos, verbos e advérbios) do que a ambiguidade de palavras funcionais (preposições, conjunções, pronomes, etc.). Em nossa experiência de anotação sintática e morfossintática de córpus, no projeto POeTiSA[1], a palavra funcional que mais apresentou ambiguidade foi o *que*. Em seus mais variados usos, o *que* se enquadra em diversas classes morfossintáticas. Embora

[1] https://sites.google.com/icmc.usp.br/poetisa

nos usos mais frequentes do *que* a anotação alcance bastante concordância entre anotadores, em seus usos menos frequentes, mas não raros, as dúvidas proliferam.

Com o intuito de debater a policategorização da palavra *que* e subsidiar decisões acerca de sua anotação dentro do esquema Universal Dependencies [1] (doravante, UD), realizamos a pesquisa aqui reportada. Organizamos o artigo em quatro seções além desta introdução. Na Seção 2 fazemos uma revisão crítica da classificação do *que* por diversos autores. Na Seção 3 discutimos a anotação do *que* na UD, tanto no nível morfossintático quanto sintático, apresentando decisões de projeto tomadas. Na Seção 4 tecemos considerações finais e delineamos possibilidades de trabalhos futuros.

2 Revisão crítica da literatura

A etimologia da palavra *que* no português é evocada por alguns gramáticos e dicionaristas para justificar diferentes classificações. De fato, a origem da palavra explica sua ambiguidade, pois o *que* representa a confluência da evolução de diferentes palavras latinas, com diferentes funções [2, 3]. Diversas gramáticas [4, 5, 6, 7, 8, 9, 10, 11] e dicionários [12, 13, 14, 15, 16] apresentam exemplos de diferentes funções da palavra *que*, mas nem sempre classificam da mesma forma os mesmos fenômenos. As gramáticas tendem a se concentrar na análise detalhada dos usos mais recorrentes, já os dicionários são excelente fonte para encontrar diversidade de usos.

Que - pronome indefinido, pronome relativo, conjunção subordinativa. São de consenso entre os autores de todas as gramáticas e dicionários acima citados as classificações do *que* como: *pronome indefinido* (1), *pronome relativo*, introduzindo orações adjetivas (2), e como *conjunção subordinativa*[2], introduzindo orações substantivas (3) e orações adverbiais (4).

(1) *Por **que** parou*?
(2) *O livro **que** li ontem é ótimo.*
(3) *É óbvio **que** isso está errado.*
(4) *Ainda **que** chova, vamos viajar.*

Como esses usos do *que* são frequentes e sua classificação não apresenta diferença nas obras consultadas (muitas obras só apresentam esses usos), não vamos estender sua discussão[3]. O que nos interessa aqui é explorar usos menos frequentes da palavra *que* (muitos deles mais frequentes na língua falada), nem sempre mencionados pelos estudiosos da língua e sobre os quais recaem algumas dúvidas de classificação.

[2] Também chamada de "conjunção integrante" quando a oração subordinada complementa a estrutura argumental de um verbo ou de um nome predicativo.

[3] Por esse mesmo motivo, não vamos usar o espaço deste artigo para discutir o uso da palavra quê, acentuada, como substantivo: "Meu bem-querer tem um quê de pecado." (verso da canção Meu Bem-Querer, de Djavan).

Que - conjunção coordenativa. Uma classificação que encontramos em dicionários [13, 14, 15, 16], mas não em gramáticas, é a do *que* como conjunção coordenativa: aditiva (5), alternativa (6), explicativa (7) e adversativa (8).

(5) *Procura **que** procura até que acha.*

(6) *"Venha **que** não venha, iniciaremos os trabalhos"* (em [15, 16]

(7) *Sai da frente **que** atrás vem gente.*

(8) "*Confie a criança a outra babá **que** não ela*." (em [15])

No exemplo 8, contudo, parece mais provável se tratar de um pronome relativo introduzindo oração com um verbo de cópula elíptico (9) do que uma conjunção equivalente à conjunção *mas* (10).

(9) *Confie a criança a outra babá **que** não [seja] ela*.

(10) **Confie a criança a outra babá **mas** não ela*.

Que - preposição. Outro uso pouco reconhecido da palavra *que* entre os gramáticos é como preposição. Os dicionaristas [13, 14, 15, 16, 17] defendem que o *que* é substituível pela preposição *exceto* em exemplos como:

(11) "*Não queirais dos livros outra unidade **que** a do seu espírito.*" (em [13])

(12) "*Não tem outros afazeres **que** os domésticos*." (em [15])

(13) "*Não podia ser outro **que** não o Padinha*." (em [14])

A substituição do *que* por *exceto* é bem plausível, contudo, não parece ser aceitável no exemplo (13), devido à presença do advérbio *não*, como mostrado a seguir:

(14) **Não podia ser outro **exceto** não o Padinha*.

Na verdade, nesse caso o *que* parece ter função de pronome relativo que introduz uma oração com verbo de cópula elíptico, igual ao que comentamos acerca do exemplo (8) anteriormente. Nessa hipótese, a sentença sem a elipse seria:

(15) *Não podia ser outro **que** não fosse o Padinha.*

O *que* é também classificado como preposição por [13, 14, 15] nos casos em que alterna com a preposição *de* como mostrado nos exemplos a seguir.

(16) *A reunião foi interrompida nada menos **que/de** três vezes*.

(17) *Você tem **que/de** entender isso!*

(18) *Há **que/de** se considerar os riscos de contágio*.

É muito conveniente distinguir a conjunção subordinativa *que*, que introduz oração finita, da preposição *que*, que introduz uma oração não finita. Como afirma [10, p.209)], "Conjunção *que* e infinitivo se excluem mutuamente".

Que - interjeição. O *que* é classificado como interjeição por [13, 14], os quais apresentam os exemplos a seguir:

(19) "***Quê** ! você por aqui?*" [13]

(20) "*Mas, **quê**! o negro estava jurado*" [14]

(21) "*Pra que se rebaixá?/Rebaixá o **quê**!*" [14]

Para [14], o *que* como interjeição tem sentido negativo, como observado nos exemplos (20) e (21). Além disso, para distinguir esse uso daquele do pronome indefinido interrogativo (*Quê?*), é muito importante observar o tipo de pontuação que o acompanha: somente o ponto de exclamação está associado à interjeição.

Como veremos a seguir, [6] chama de interjeição o uso que [13, 14] chamam de pronome exclamativo.

Que - pronome exclamativo. O uso do *que* classificado por [6] no exemplo (22) como interjeição, é classificado por [13, 14] como pronome exclamativo:

(22) "*Eugênia sentou-se a concertar uma das tranças.* ***Que*** *dissimulação graciosa!* ***Que*** *arte infinita e delicada!* ***Que*** *tartufice profunda!*" [6, p. 107]

(23) ***Que*** *paisagem!* ***Que*** *paisagem linda! Olha só* ***que*** *linda paisagem!*

É claro, porém, que se trata do mesmo fenômeno, ou seja, um modificador nominal que confere um caráter exclamativo ao enunciado, muitas vezes em frases sem verbos.

Que - advérbio. [9] já mencionava o *que* como advérbio, argumentando que vinha do latim *quam* (quão = muito), ideia que é compartilhada por [12, 13 e 14], que o classificam como advérbio de intensidade. Esse advérbio destaca-se dos demais da classe pelo fato de não modificar verbos, apenas adjetivos e alguns advérbios. Outra particularidade é o fato de que esse advérbio tem uma função similar à do pronome exclamativo *que*, modificador de substantivos:

(24) ***Que*** *lindo foi o espetáculo que nós vimos!* (que = advérbio)

(25) ***Que*** *lindo!* (que = advérbio)

(26) ***Que*** *lindo espetáculo nós vimos!* (que = pronome exclamativo)

A distinção entre advérbio de intensidade e pronome exclamativo, contudo, é conveniente pelo fato de essa similaridade não ser observada em outras línguas, como francês e inglês, que possuem formas diferentes para as duas funções.

Que - função de focalização. Dois estudos sobre funções pragmáticas no português [18, 19] trazem subsídios para classificar outras ocorrências do *que*. Dentro das funções pragmáticas, tanto o *que* isoladamente quanto o *que* precedido do verbo *ser* podem ser utilizados para focalizar um constituinte da oração:

(27) *É de oportunidades de trabalho* ***que*** *precisamos.*

(28) *Só depois é* ***que*** *percebi que havia esquecido a carteira.*

(29) *Desde cedo* ***que*** *não como nada.*

(30) *Há anos* ***que*** *não nos vemos.*

[19] refere-se a esse *que* como conjunção subordinativa, mesmo nos períodos não compostos por subordinação. Na verdade, esse *que* não encontra classificação morfossintática mais adequada que essa nas classes tradicionais da gramática do português, principalmente porque, sintaticamente, não tem função, podendo ser suprimido sem prejuízo para a gramaticalidade da sentença.

Que - expletivo. [17] aponta o caso do *que* iniciando orações com o verbo no subjuntivo e classifica-o como expletivo por poder ser suprimido e não ter função de focalização. Esse *que* é, na verdade, conjunção subordinativa que "sobrou" após a elipse de um verbo desiderativo na oração matriz: *Espero que, Desejo que*.

(31) ***Que*** *todos façam uma boa viagem!*

Que - não classificado. Há outros usos em que o *que* expressa um sentido negativo (talvez por efeito de ironia), similar ao apontado por [14] nos exemplos (20) e (21), porém sempre associado a substantivos, adjetivos ou verbos.

(32) ***Que*** *cristão,* ***que*** *nada!*

(33) ***Que*** *pescar,* ***que*** *nada!*

A classificação do *que* nessas construções não consta das obras consultadas e não nos parece óbvia. Pelo fato de apresentar semelhança com os casos classificados por [13, 14] como interjeição, inclusive atribuindo um caráter negativo ao termo ao qual se associa, parece-nos apropriado classificar esse uso como interjeição.

3 A anotação do *que* na UD

A UD[4] [1] é uma iniciativa multinacional de anotação de córpus com relações de dependência. O objetivo é tornar o esquema de anotação o mais genérico possível para que possa ser aplicado a diversas línguas. Por essa razão, antes de adotar esse esquema em um projeto de anotação, é preciso que as diretrizes da UD sejam instanciadas para descrever como elas se aplicam à língua em foco.

A UD possui um conjunto de 17 etiquetas morfossintáticas (PoS tags) e outro com 37 relações de dependência (chamadas deprel, de ***dep****endence* ***rel****ations*). Diretrizes para o uso dessas etiquetas em português são apresentadas em [20, 21]. Essas diretrizes estão sendo utilizadas na anotação do córpus multigênero Porttinari [22].

O primeiro córpus anotado dentro do Porttinari, contendo 168.397 tokens, apresentou 3.488 ocorrências do *que*, 1.880 das quais anotadas como PRON (pronome) e 1.579 como SCONJ (conjunção subordinativa). Apenas 29 casos eram de outras categorias, número que atribuímos ao fato de o gênero desse primeiro córpus ser jornalístico.

O que diferencia o *que* PRON do *que* SCONJ é o fato de apenas o primeiro ter um papel sintático e, dependendo desse papel, o *que* PRON participa de diferentes deprel. A Figura 1 ilustra um caso no qual o *que* PRON é dependente da deprel **obl** (*oblique*), classificação dos objetos indiretos na UD. No exemplo da Figura 1, por estar precedido de preposição, o *que* participa de mais uma relação de dependência: a relação **case**, na qual ele é *head* e a preposição *com* é dependente.

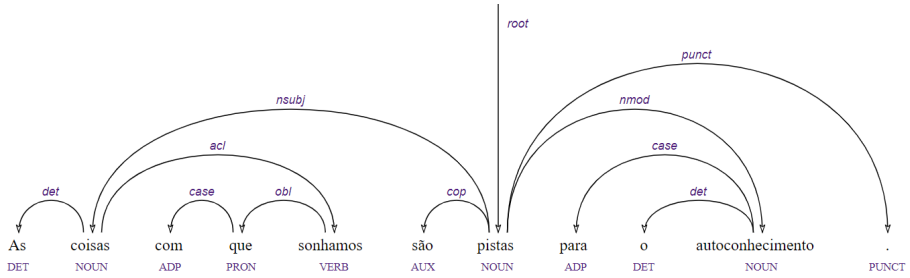

Fig. 1. Anotação do *que* pronome relativo.

[4] https://universaldependencies.org/

Já o *que* SCONJ é, quase sempre, dependente da relação **mark**, que liga o *que* ao predicado da oração subordinada, como ilustrado na Figura 2.

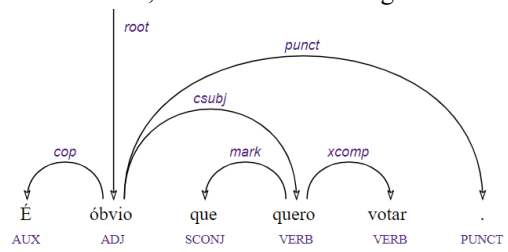

Fig. 2. Anotação do *que* conjunção subordinativa

Uma exceção é quando o *que* é acompanhado de outras palavras que, com ele, constituem uma expressão fixa. Nesses casos, a relação de dependência utilizada é a **fixed**, em que o *head* é a primeira palavra da expressão e os dependentes são as demais palavras (como a locução temporal *assim que* na Figura 3).

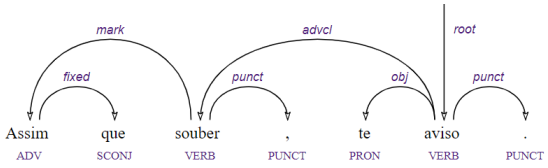

Fig. 3. Anotação do *que* como parte de uma locução conjuntiva subordinativa

Há várias locuções conjuntivas subordinativas nas quais o *que* é anotado como SCONJ e faz parte de uma relação **fixed**: *assim que, logo que, desde que, sempre que, nem que, uma vez que, à medida que*, *a menos que, a não ser que, para que, a fim de que, que nem, etc.* [23].

Também são anotadas com a relação **fixed** as locuções coordenativas *só que* e *ao passo que* (substituíveis por *mas*) nas quais o *que* é anotado como CCONJ. E, por fim, temos dois casos em que o *que* é PRON e também participa de uma deprel **fixed**: quando precedido do demonstrativo *o* (*o que*), em que o *o* pode ser suprimido sem prejuízo para a gramaticalidade, e na expressão comparativa *do que* (*de o que*).

É importante esclarecer que uma mesma sequência de palavras pode constituir uma locução em um contexto, mas não em outro. Isso impede que expressões **fixed** contendo o *que* sejam anotadas automaticamente, sem revisão humana:

(34) *Ele dorme **que nem** um anjo.*
(35) *Ele dorme tão profundamente **que nem** um trovão o acordaria.*
(36) *Não sei **o que** fizeram com você*
(37) *O professor avisou-**o**[5] **que** seria punido.*

[5] Com a tokenização, o hífen que separa os clíticos é eliminado, o que torna o pronome oblíquo *o* sem marca que o distinga do pronome demonstrativo *o*.

Outro fato digno de nota é que, nas construções comparativas, é comum o verbo e até o termo comparado estarem elípticos, mas isso não deve comprometer a classificação da oração adverbial comparativa, conforme ilustrado pela Figura 4. Sem elipse, a sentença seria: *Seus argumentos são mais consistentes que os meus argumentos são*.

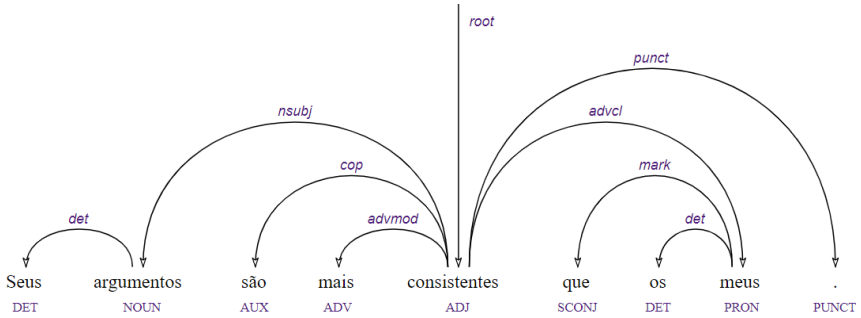

Fig. 4. Anotação do *que*, conjunção comparativa, como SCONJ

Outro caso comum de elipse que observamos em nosso córpus é a do verbo de cópula em orações adverbiais concessivas com predicado nominal:

(38) *Ainda **que** doente, foi trabalhar.* (=*Ainda **que** estivesse doente…*)

O *que* exclamativo é anotado com DET (etiqueta morfossintática na qual a UD reúne todos os artigos e pronomes que modificam substantivos), e com a deprel **det**.

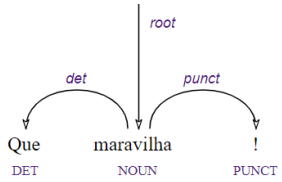

Fig. 5 Anotação do *que*, pronome exclamativo, como DET

O *que* advérbio, por sua vez, foi anotado como ADV e com a deprel **advmod**:

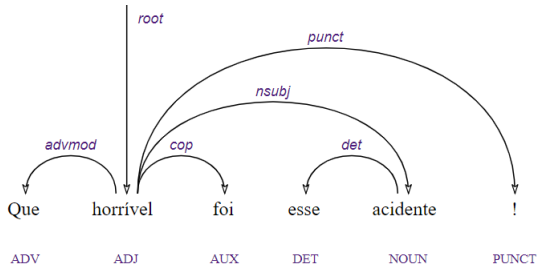

Fig. 6 Anotação do *que*, advérbio de intensidade, com ADV

Construções de focalização usando *que* foram anotadas de forma semelhante à descrita por [24] e adotada na anotação do Bosque-UD [25], ou seja, o *que* foi anotado com a etiqueta SCONJ no nível morfossintático e participa como dependente

da relação **discourse**. Outra alternativa seria a deprel **expletive**, mas não vimos motivo para divergir da anotação descrita no trabalho citado, uma vez que se trata de uma função pragmática para a qual não temos uma relação sintática convencional.

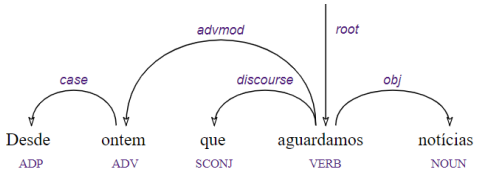

Fig. 8 Anotação do *que* com função de focalizador

A Tabela 1 traz exemplos dos casos discutidos, sua classificação pelos autores consultados e sua anotação no esquema UD.

4 Considerações Finais

Embora tenhamos levantado tipos variados de ocorrências do *que*, nossa experiência nos mostrou que a quantidade de usos em córpus sempre supera o esforço de prevê-los em qualquer tipo de manual, gramática ou dicionário, principalmente por influência dos gêneros. Sendo assim, o trabalho aqui relatado será sempre estendido à medida que novos usos forem atestados.

Por falta de espaço, não ilustramos a anotação de todos os usos levantados do *que*, mas criamos um projeto de anotação contendo pouco mais de 100 sentenças com os diferentes usos do *que* identificados, disponibilizado em ambiente de anotação[6]. Essas sentenças ilustram o uso do *que* aqui discutidos, bem como casos de expressões recorrentes, como: *Tenta,* ***vai que*** *dá certo*; ***Até que*** *eu topava, se me convidassem*; *O remédio* ***fez***[7] ***com que*** *sarasse*; ***Será que*** *funciona?*

Tais sentenças poderão ser usadas para: 1) conhecer a grande diversidade de usos do *que*, sem repetição; 2) conhecer casos difíceis de anotação do *que*, que poderão ser discutidos pelos grupos que anotam córpus utilizando a abordagem UD; 3) pesquisar a ocorrência de alguns tipos de construções em córpus e, se necessário ou desejável, promover artificialmente o aumento de ocorrências semelhantes a fim de diminuir o efeito negativo da esparsidade de dados sobre o aprendizado automático.

Esperamos que os casos aqui discutidos sirvam, ao menos em parte, para abreviar o esforço de anotação de todos aqueles que se dedicam a anotar córpus de português seguindo a abordagem da UD e suscitem discussões, em especial sobre o uso sintaticamente "opaco" do *que* em funções pragmáticas.

---

[6] https://arborator.icmc.usp.br/#/projects/Anotação_do_"QUE"

[7] uso causativo do verbo *fazer*

Tabela 1. Usos do *que* e suas respectivas classificações na literatura e na UD

Uso	Classificação na literatura	UD		
		POS	DEPREL das quais participa	
			head	dependente
Você tem **que** entender isso!	preposição	ADP	-	mark
Isso ocorreu em **menos que** três ensaios.	preposição	ADP	-	fixed
Que linda eu era!	advérbio	ADV	-	advmod
Que nada!	-	INTJ	-	discourse
Sai da frente **que** atrás vem gente.	conjunção	CCONJ	-	cc
A **que** período você se refere?	pronome interrogativo	DET	-	det
Que espetáculo!	pronome exclamativo	DET	-	det
Que, não liga pra isso não.	interjeição	INTJ	-	discourse
Por **que** parou?	pronome	PRON	case	obl
O livro **que** li ontem é ótimo.	pronome	PRON	-	obj
O que você quer?	pronome indefinido	PRON	-	fixed
É óbvio **que** isso está errado.	conj. subordinativa	SCONJ	-	mark
Ainda que chova, vamos viajar.	conj. subordinativa	SCONJ	-	fixed
Ele dorme **que nem** um anjo.	conj. subordinativa	SCONJ	fixed	mark
Só depois **que** vi isso.	função de focalização	SCONJ	-	discourse
Que todos façam uma boa viagem!	expletivo	SCONJ	-	expletive
Sofreu tanto **que** desistiu de viver.	conj. subordinativa	SCONJ	-	mark

Agradecimentos

As autoras agradecem o apoio do Centro de Inteligência Artificial da Universidade de São Paulo (C4AI-http://c4ai.inova.usp.br/), financiado pela IBM e pela FAPESP (processo#2019/07665-4).

Referências

1. Nivre, J.; Marneffe, M. C.; Ginter, F.; Hajič, J.; Manning, C. D.; Pyysalo, S.; Schuster, S.; Tyers, F.; Zeman, D.: Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection. In: Proceedings of the 12nd LREC, p. 4034-4043 (2020).
2. Gonçalves, L. M. M.: O complementizador latino Quod. In: Rodrigues, Ângela C. de S. (eds.) 50º Seminário do GEL 2002, vol. 32. Estudos Linguísticos, São Paulo (2003).
3. Almeida, N. M. de: Gramática latina: curso único e completo. 29ª ed. Saraiva, São Paulo (2000).
4. Azeredo, J. C. de: Fundamentos de Gramática do Português. 3.ed. Jorge Zahar, Rio de Janeiro (2000a).
5. Azeredo, J. C. de: Iniciação à Sintaxe do Português. 8ª ed. Jorge Zahar, Rio de Janeiro (2000b).
6. Bechara, E.: Moderna gramática portuguesa 37ª ed. Nova Fronteira, Rio de Janeiro (2009).
7. Castilho, A. T.: Nova Gramática do Português Brasileiro. São Paulo, Editora Contexto. (2010)
8. Cunha, C.; Cintra, L.: Nova Gramática do Português Contemporâneo. 2nd edn. Lexicon, Rio de Janeiro (2017).
9. Neves, M. H. M. Gramática de Usos do Português. Editora UNESP (1999).
10. Perini, M. A.: Gramática Descritiva do Português Brasileiro. Vozes,Petrópolis (2016).
11. Rocha-Lima.: Gramática Normativa da Língua Portuguesa. 49ª ed. José Olympio, Rio de Janeiro (2011).
12. Figueiredo, C. de: Novo Diccionário da Língua Portuguesa (1913) Domínio Público.
13. Ferreira, A. B.: Novo Dicionário da Língua Portuguesa. 2ª ed. Nova Fronteira, Rio de Janeiro (1986).
14. Borba, F. S.: Dicionário de usos do Português do Brasil. 1ª ed. Editora Ática, São Paulo (2002).
15. Dicionário de Português da Google. Oxford Languages. Obra on line: https://languages.oup.com/google-dictionary-pt/
16. Grande Dicionário Houaiss. Obra on line: https://houaiss.uol.com.br
17. Michaelis Moderno Dicionário da Língua Portuguesa. Editora Melhoramentos Ltda. (2022). Obra on line: https://michaelis.uol.com.br/moderno-portugues/
18. Longhin, S. R.; Ilari, R. Uma leitura hallidayiana das sentenças clivadas do português. ALFA: Revista de Linguística, vol. 44, São Paulo (2001). Disponível em: https://periodicos.fclar.unesp.br/alfa/article/view/4205.
19. Pezatti, E. G.: Clivagem e construções similares: contraste, foco e ênfase. Linguística, v. 28, p. 73-98, (2012). Disponível em: <http://hdl.handle.net/11449/122327>.
20. Duran, M.S. Manual de Anotação de PoS tags: Orientações para anotação de etiquetas morfossintáticas em Língua Portuguesa, seguindo as diretrizes da abordagem Universal Dependencies (UD). Relatório Técnico do ICMC 434. Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo. São Carlos-SP, Setembro, 55p. (2021).
21. Duran, M.S. Manual de Anotação de Relações de Dependência: Orientações para anotação de relações de dependência sintática em Língua Portuguesa, seguindo as diretrizes da abordagem Universal Dependencies (UD). Relatório Técnico do ICMC 435. Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo. São Carlos-SP, Dezembro, 79p. (2021).

22. Pardo, T.A.S.; Duran, M.S.; Lopes, L.; Di Felippo, A.; Roman, N.T.; Nunes, M.G.V. Porttinari - A large multi-genre treebank for Brazilian Portuguese. In the Proceedings of the XIV Symposium in Information and Human Language (STIL), pp. 1-10. November, 29 to December, 3. (2021).
23. Oliveira, T. P.: Conjunções Adverbiais no Português. Revista de Estudos da Linguagem, vol.22, nr. 1 (2014).
24. Souza, E.; Cavalcanti, T.; Silveira, A.; Evelyn, W.; Freitas, C.: Diretivas e documentação de anotação UD em português (e para língua portuguesa). (2020). Disponível em: https://nbviewer.jupyter.org/github/comcorhd/Documenta-o-UD-PT/raw/master/Documenta-o-UD-PT.pdf
25. Rademaker, A.; Chalub, F.; Real, L.; Freitas, C.; Bick, E.; Paiva, V. Universal Dependencies for Portuguese. In: Proceedings of the Fourth DEPLING, p. 197-206. Pisa, Itália, Linköping University Electronic Press (2017).

Shallow parsing of Portuguese texts annotated under Universal Dependencies

Guilherme Martiniano de Oliveira[0000−0002−2030−3688], Paulo Berlanga Neto[0000−0002−1985−1089], Evandro Eduardo Seron Ruiz[0000−0002−7434−897X]

Department of Computing and Mathematics – FFCLRP
University of São Paulo, Ribeirão Preto, SP – Brazil
{guizera11, pauloberlanga, evandro}@usp.br

Abstract. Shallow parsing is an intermediate step to many natural language processing tasks, such as information retrieval, question answering, and information extraction. An alternative to full-sentence parsing consists of segmentation and identifying phrases in sentences. Building such a parser for the Portuguese language is challenging considering the proposed formalism for grammar annotation, the Universal Dependency (UD). This paper addresses preliminary studies to overcome these barriers by annotating noun phrases tagged in UD.

Keywords: Shallow parsing · Universal Dependencies · Neural Networks

1 Introduction

Assigning a complete syntactic structure to sentences based on grammar and a search strategy is the goal of full parsing. However, not all-natural language processing (NLP) applications require a complete syntactic analysis [9]. For many NLP tasks, such as named entity recognition [3], sentiment analysis [15] and information retrieval [6], recovering only a limited amount of syntactic information has proved to be a valuable technology for written and spoken language domains. This chunking strategy is generally known as partial parsing or shallow parsing. Shallow parsing can also serve as a baseline for full parsing [2] since it provides a foundation for other levels of analysis.

This work focuses on extracting non-overlapping noun-phrase (NP) chunks, as proposed initially by Abney [2], including nouns and proper nouns, among other classes of words that add more meaning to these two. Shallow parsers have already been developed for the constituency tree format [5]. Here we address the challenge of developing a parser to work under Brazilian-Portuguese texts annotated with the Universal Dependencies (UD) format, which is currently used in many NLP tasks.

We propose constructing a model for recognizing noun phrases in input sentences through a neural network (NN) trained model, as proposed by Søgaard and Goldberg [16]. Some NN architectures are explicitly designed for long-term

dependency learning, as written texts are. More specifically, our proposed shallow parser model processes text in three stages: 1) A *learning corpus* is built from partial parsed sentences. These sentences are extracted from the constituency version of Bosque corpus (version 8); 2) Sentences from this learning corpora are augmented with UD labels from the UD_Portuguese-Bosque version 2.2. This revised UD treebank retains the additional tags for NP. Finally; 3) A neural network-based classification model is built from the learning corpus and applied to the original test subset from the UD_Portuguese-Bosque, here called *text corpus*.

Following, we briefly introduce the main related work to shallow parsing. In Section 3, we present the data and methods used. In Section 4, we report a summary of the experiments. Some considerations about the experiment's results are detailed in Section 5. Finally, in Section 6, we present some concluding remarks.

2 Related work

The idea of text chunking was proposed in the seminal work of Steven Abney [2], where he shows the correspondence of prosodic patterns to segments of constituency grammar trees. Following this intuition, Ramshaw and Marcus [14] developed the first known method for chunking sentences similarly to traditional grammar, creating templates and rules that described chunk formation. This method is known as Transformation-Based Learning (TBL).

Alonso et al. [3], Brants [6] and a team led by Hammerton [9], among others, have also developed and applied shallow parsing to sentences annotated in the constituency tree format.

For the Portuguese language, we highlight the work of Barreto and his colleagues [4] with the TagShare project that embraces linguistic resources and tools for the shallow processing of Portuguese. These resources also include a 1M token corpus that has been accurately hand-annotated. Noun phrase chunking for English, Portuguese, and Hindi was proposed by Milidiú, Santos, and Duarte [12]. They applied Entropy Guided Transformation Learning (ETL), a machine learning strategy that combines decision trees and the classical TBL method. For the Portuguese, their proposed methodology achieved a precision of 92.62%, recall of 93.05%, and an F-measure of 92.84%.

Machine learned-based system was also used for a shallow parsing similar task called clause identification (CI). The Milidiú team extended their previous experiments to work likewise with CI [8]. They stated that CI is a phrase-chunk-like (PCL) task. PCL consists of splitting a sentence into clauses. A clause is defined as a word sequence containing a subject and a predicate. Clause identification is a special kind of shallow parsing. They proposed an Entropy Guided Transformation Learning system that achieved an F-measure of 73.9%.

Chunking received much attention, mostly when syntactic parsing was predominantly guided by constituency parsing, as it is the case for all previous works. With the UD grammar annotation surge, new methods need to be cre-

ated. To our knowledge, Ophélie Lacroix [11] was the first to show that UD annotated texts can also leverage the information provided by the constituency annotation. She grouped tokens to form NP chunks and used neural networks to train and test her method. She showed that it is possible to extract NP-chunks (noun phrases) from Universal Dependencies annotated texts with accuracy similar to traditional chunks operated under constituency trees. Her NP-chunking method achieved F-measure=89.9% when applied to dependency trees.

Our project aims to deduce NP-chunks from automatically UD annotated texts using a deep neural network (NN) approach. To lead our way to a feasible NN model for NP-chunking, we based our project on the work of Søgaard and Goldberg [16]. They showed that it is possible to utilize a multi-task learning architecture (MTL) with deep bi-directional recurrent neural networks (RNNs) to make syntactic chunking more precise, achieving an F-score=94.1%. They conclude that deep neural networks are a powerful tool for syntactic analysis.

3 Methodology

3.1 Data

Using an NN-trained model, we aim to recognize and extract non-overlapping noun phrase (NP) chunks. As requested by a supervised learning approach, two corpora are needed: a) A *learning corpora*, and; b) A *test corpora*. The *learning corpora* used is composed of sentences from the Bosque corpus. Version 8.0 of the Bosque corpus[1] provides syntactic annotations of noun phrase chunks, under the 'NP' category, like other types of phrase chunks. As a constituency parsed corpus, no UD labels were provided for this version of the Bosque. UD labels were acquired from the UD_Portuguese-Bosque version 2.2[2]. This UD treebank retained the original NP tags. The *test corpora* is composed of the test subset labeled sentences from the UD_Portuguese-Bosque.

A classification engine (detailed in the next subsection, 3.2) is fed with the *test corpora* sentences. Each extracted sentence is analyzed accordingly to the knowledge acquired from the *learning corpora*. The following subsection describes the classification engine.

3.2 Method

We define the noun phrase detection task as a sequence labeling problem. Given an input sentence composed of a sequence of tokens, $w_1, \ldots, w_n$, the goal is the prediction of an output sequence $y_1, \ldots, y_n$, $y_i \in \{1, \ldots, |L|\}$, where L is a determined set of labels and y_i is the respective label for w_i.

We adopted an MTL architecture based on deep bi-directional recurrent neural networks (Bi-LSTM). The MTL can be understood as a layer-sharing method

[1] https://www.linguateca.pt/Floresta/corpus.html#download
[2] https://github.com/UniversalDependencies/UD_Portuguese-Bosque

that helps models deal with different tasks simultaneously. Therefore, such intermediary representations allow different tasks to benefit from each other, stimulating the standard practical knowledge learning process. Considering the proposed sequence labeling model, we may, for example, experiment with part-of-speech (POS) tagging and syntactic chunking predictions for the same input sentence.

Long Short-Term Memory (LSTM) [10] is a particular flavor of recurrent neural networks (RNN) widely applied in NLP tasks that enables long-term dependency learning. It may also be considered an instance that primarily aims to eliminate the vanishing gradient problem observed in the 'vanilla' RNN [7] since the latter cannot correctly handle long sequences of tokens [16].

Explained in a simple way, the LSTM architecture, consider RNNs as a black-box abstraction. One may view LSTMs as an instance of a RNN interface. RNN may be seen as a function $R_{\Theta}(w_{1:n})$ mapping a sequence of n input vectors $w_{1:n}, w_i \in R_{\text{in}}$, to output vector $h_{1:n}, h_i \in R_{\text{out}}$. Applying $R_{\Theta}(w_{1:n})$ to all prefixes $w_{1:i}, 1 \leq i \leq n$ of $w_{1:n}$, result in n output vectors $h_{1:n}$, where $h_{1:i}$ is a summary of $w_{1:i}$.

Layers of RNN are called deep RNN. A $k-$layer RNN are a set of k RNN functions $(\text{RNN}_1, \text{RNN}_2, \ldots, \text{RNN}_k)$ feeding each other. A bidirectional RNN is composed of two RNNs, RNN_{F} and RNN_{R}, one that reads the sequence in one order, e.g., forward, and the other reading it in reverse.

We employed an architecture-based Bi-LSTM following Søgaard and Goldberg reference work [16]. They show that this architecture can explore contextual information to process long sequences. Our proposed model comprises an embedding layer that feeds two hidden layers (forward and backward), composed of 300 units. The model was trained using back-propagation and Stochastic Gradient Descent (SGD), employing batch sizes of 64 with a learning rate of 0.01. The training process lasted ten epochs. All the hyper-parameters were defined empirically. The Bi-LSTM implementation was accomplished with the `nlp-architecture`[3] Python module [1].

4 Experiment

We recall that although the Bosque corpus version 8 is composed of 18,804 sentences, only part of this corpus, 9,364 sentences, were annotated under UD, assembling the UD_Portuguese-Bosque. Further, these 9,364 sentences are divided into three subsets: learning-train (8,328), dev (560) and test 476. Since not all the sentences have NP and some processing errors, such as bugs reading the XML file, only 8,585 sentences were used, corresponding to 91,6% of the 9,364. Table 1 below depicts the number of sentences used for both corpora, the *learning* (7,605) and the *test* (444) corpus.

Based on the syntactic annotations provided by the Bosque corpus (v.8), we acknowledge noun phrase chunks searching for tokens inside the noun phrase (*'NP'* category) also considering the alongside adjectives (*'adpj'* category). Figure 1(a) illustrates such annotations in the Bosque `SimTreeML` format.

[3] https://intellabs.github.io/nlp-architect/

Table 1. Number of sentences for the used corpora.

subset	Bosque v.8 (constituency)	UD_Portuguese-Bosque (original)
learning-train	**7,605**	8,328
dev	536	560
test	**444**	476
Total	8,585	9,364

Fig. 1. Steps performed to the annotation process.

After that, we annotate each token from every sentence with the respective labels from the Universal Dependencies (UD) annotation format. In parallel, NP chunks were labeled with the IOB (Inside–Outside–Beginning) format [14]. Figure 1(b) illustrates the final annotated example.

Following the work of Lacroix [11], we aim to detect minimal, non-recursive noun phrases. For example, in the sentence *"O 7 e Meio é um ex-libris da noite algarvia."*, we consider the following constituents: *"O 7 e Meio"*, *"um ex-libris"* and *"a noite algarvia."*. Thus, we do not consider a single long noun phrase for *"um ex-libris da noite algarvia."*, but the aforesaid minimal version instead.

4.1 Evaluation

We assembled the Bosque data division in train-development-test subsets according to the work of Rademaker et al. [13]. See Table 1. Later, we trained the model with the previously mentioned method in Section 3.2. Running the

test against the full reserved test set, we obtained an F-measure of 85.1%. See Table 2.

Table 2. Evaluation metrics for the Bi-LSTM network model in %.

Precision	Recall	F-measure
84.8	85.3	85.1

We may also see in Figure 2 an example of a prediction outputted by the trained model that correctly identifies the noun phrases present in the input sentence provided, based on the IOB pattern.

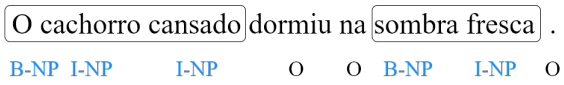

Fig. 2. Noun phrase prediction produced by the proposed model.

5 Considerations

A rudimentary qualitative analysis of the outputs reveals that the model could detect the desired minimal noun phrase chunks performing slightly better on sentences with simple syntax. Even so, many of the longest and most complex sentences were also labeled correctly. Quantitatively, an F-measure of 85.1% is not a state-of-the-art achievement. Although this work is not comparable with the work of Lacroix [11] that achieved an F-measure of 89.9%, we considered our result an encouraging preliminary one. The Bi-LSTM classifier was used with its default parameters, suggesting that an optimized gradient boosting approach like XGBoost would provide more gratifying results. The obtained F-score establishes our approach as a feasible method for Portuguese text chunking.

Although a comprehensively qualitative manual inspection of the errors shall be the subject of a prospective study, a casual manual search for minimal NP reveals some inconsistencies in the original POS tagging. Below we highlight the expression "(P)presidente da (R)república", which should not be tagged as a minimal NP. One may see a possible disagreement between human annotators in the following expressions.

1. . . . o governador do Rio e o **Presidente**[PROPN] da **República**[PROPN] chamaram o Exército.
2. . . . o **presidente**[NOUN] da **República**[NOUN] abriu uma fresta . . .
3. No caso de impedimento de o **presidente**[NOUN] da **República**[PROPN] . . .

In the previous examples the word "Presidente", with a capital 'P' is tagged as *Proper Noun* while "presidente" is tagged as *Noun*. Respectively "República" appears with two distinctive tags, *Proper Noun* and *Noun*. Originally, the expression in the first sentence, "Presidente da República" was tagged as a NP, while for the second and third sentences, "presidente" and "República" were tagged individually as NP.

These last divergent examples encourage an extensive investigation, even if insufficient to justify the modest F-measure obtained. We note that the learning step might be impaired on comparable divergences due to the relatively small training dataset for the enormous variety of similar expressions.

6 Conclusion

We inferred that the method proposed has much potential for chunking detection that takes advantage of the characteristics presented in the UD pattern. We also believe that expanding learning corpora annotated under UD will foster more encouraging results. An accuracy over 95% and new methods to extract other types of chunks (prepositional, adverbial, and adjective) are some future works we are pursuing.

7 Acknowledgements

This work was carried out at the Center for Artificial Intelligence (C4AI-USP), with support by the São Paulo Research Foundation (FAPESP grant #2019/07665-4) and by the IBM Corporation.

References

1. NLP Architect, by Intel AI Laboratories (Nov 2018), https://doi.org/10.5281/zenodo.1477518
2. Abney, S.P.: Principle-Based Parsing: Computation and Psycholinguistics, chap. Parsing by Chunks, pp. 257–278. Springer Netherlands, Dordrecht (1992). https://doi.org/10.1007/978-94-011-3474-3_10
3. Alonso, M.A., Gómez-Rodríguez, C., Vilares, J.: On the Use of Parsing for Named Entity Recognition. Applied Sciences **11**(3), 1090 (2021)
4. Barreto, F., Branco, A., Ferreira, E., Mendes, A., Bacelar do Nascimento, M.F., Nunes, F., Silva, J.R.: Open resources and tools for the shallow processing of Portuguese: the TagShare project. In: Proceedings of the V International Conference on Language Resources and Evaluation – LREC2006. European Language Resources Association (2006)
5. Bick, E.: The Parsing System "Palavras": Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework. Aarhus University Press (2000), https://books.google.com.br/books?id=ISUgDvPg7hcC
6. Brants, T.: Natural Language Processing in Information Retrieval. CLIN – Computational Linguistics in the Netherlands **111** (2003)

7. Elman, J.L.: Finding structure in time. Cognitive Science **14**(2), 179–211 (1990)
8. Fernandes, E.R., dos Santos, C.N., Milidiú, R.L.: A Machine Learning Approach to Portuguese Clause Identification. In: Pardo, T.A.S., Branco, A., Klautau, A., Vieira, R., de Lima, V.L.S. (eds.) Computational Processing of the Portuguese Language. pp. 55–64. Springer Berlin Heidelberg, Berlin, Heidelberg (2010)
9. Hammerton, J., Osborne, M., Armstrong, S., Daelemans, W.: Introduction to Special Issue on Machine Learning Approaches to Shallow Parsing. Journal of Machine Learning Research **2**(4), 551–558 (2002). https://doi.org/10.1162/153244302320884533
10. Hochreiter, S., Schmidhuber, J.: Long Short-Term Memory. Neural Comput. **9**(8), 1735–1780 (Nov 1997). https://doi.org/10.1162/neco.1997.9.8.1735, https://doi.org/10.1162/neco.1997.9.8.1735
11. Lacroix, O.: Investigating NP-chunking with Universal Dependencies for English. In: Proceedings of the Second Workshop on Universal Dependencies (UDW 2018). pp. 85–90. Association for Computational Linguistics, Brussels, Belgium (Nov 2018). https://doi.org/10.18653/v1/W18-6010, https://aclanthology.org/W18-6010
12. Milidiú, R.L., dos Santos, C., Duarte, J.C.: Phrase chunking using entropy guided transformation learning. In: Proceedings of ACL-08: HLT. pp. 647–655 (2008)
13. Rademaker, A., Chalub, F., Real, L., Freitas, C., Bick, E., de Paiva, V.: Universal Dependencies for Portuguese. In: Proceedings of the Fourth International Conference on Dependency Linguistics (Depling). pp. 197–206. Pisa, Italy (September 2017), http://aclweb.org/anthology/W17-6523
14. Ramshaw, L.A., Marcus, M.P.: Text Chunking Using Transformation-Based Learning, pp. 157–176. Springer Netherlands, Dordrecht (1999). https://doi.org/10.1007/978-94-017-2390-9_10
15. Sharma, A., Gupta, S., Motlani, R., Bansal, P., Shrivastava, M., Mamidi, R., Sharma, D.M.: Shallow parsing pipeline – Hindi-English code-mixed social media text. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1340–1345. Association for Computational Linguistics, San Diego, California (Jun 2016). https://doi.org/10.18653/v1/N16-1159, https://aclanthology.org/N16-1159
16. Søgaard, A., Goldberg, Y.: Deep multi-task learning with low level tasks supervised at lower layers. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). pp. 231–235. Association for Computational Linguistics, Berlin, Germany (Aug 2016). https://doi.org/10.18653/v1/P16-2038, https://aclanthology.org/P16-2038

Still on arguments and adjuncts: the status of the indirect object and the adverbial adjunct relations in Universal Dependencies for Portuguese *

Elvis de Souza and Cláudia Freitas

Pontifical Catholic University of Rio de Janeiro
elvis.desouza99@gmail.com,
claudiafreitas@puc-rio.br

Abstract. We report the process of annotating verbal arguments and adjuncts in PetroGold, a treebank of the oil & gas domain. The corpus follows the dependencies approach of the Universal Dependencies multilingual project. The argument-adjunct distinction in UD is not a relevant one, and it is up to the contributors of each language to decide how to annotate it in some particular cases. After consulting Portuguese grammars to assist in the annotation of the adverbial adjunct and indirect object relations, we propose a semantic-discursively oriented approach, which was used in the PetroGold annotation and affected 14.8% of the sentences in the treebank. Finally, we present a visualization of the results, showing the distribution of verbs by transitivity in the corpus.

Keywords: Treebank annotation · Universal Dependencies guidelines · Portuguese grammar

1 Introduction

When syntactically annotating or revising a treebank, every word, phrase or term in a sentence must be classified. When there is not a possibility for multiple categorisation, in many cases the distinction between one class and another is not trivial. The difficulties may arise from lack of studies of the linguistic phenomenon or lack of specific annotation guidelines.

In Portuguese, the distinction between indirect objects (one of the verbal arguments) and adverbial adjuncts can be particularly difficult in some cases due to the fact that both phrases are prepositioned – in the adverbial adjunct, the difficulty only occurs when it is prepositioned – and both are dependent on the predicate head. None could say that both classes were not thoroughly studied, but the tendency of grammars is to simplify the subject, presenting prototypical sentences in which the distinction is more easily made. Our concern, however, is

* Elvis de Souza thanks the National Council for Scientific and Technological Development (CNPq) for the Masters scholarship process no. 130495/2021-2.

with real corpus sentences, such as sentences (1)-(3), found in Bosque [1], where distinguishing between verbal complement and adverbial adjunct is not a simple task.

1. Os jogadores se dividem **pelos dez quartos** do alojamento, equipados com frigobar, ar condicionado, televisão e telefone.[1]
2. Papa indica mulher **para secretaria**[2]
3. O PDT pretende reduzir os impostos federais **a quatro**.[3]

In this work, we report the process of annotating the prepositioned complements of verbs in the second version of PetroGold [9], a gold standard treebank of the oil & gas domain. The corpus contains 250,595 tokens (8,949 sentences) morphosyntactically annotated according to the multilingual annotation of the Universal Dependencies [6] project. Therefore, our starting point for studying the phenomenon is the project guidelines, discussed in section 2.1.

After noticing that the project allows each language contributors to find their own solutions to language-specific constructions such as the prepositional objects in Portuguese, we verify what Brazilian and Portuguese grammarians have said on the argument-adjunct distinction in section 2.2. Vilela and Koch [10], for instance, recognize that the argument-adjunct distinction "has deserved some reflection and a definitive conclusion has not yet been reached" (transl., p. 347). We will see inconsistencies in the criteria suggested by these and other authors as well as incentive to our proposal of a semantic-discursive criterion to differentiate prepositional objects from adverbial adjuncts.

We present our annotation proposal with the aim to increase the inter-annotator agreement without giving up meaningful linguistic information. We report the methodology used in the PetroGold annotation in section 3, and, in section 4, we carry out a study of the subcategorization of verbs in the corpus according to the results obtained.

2 A multilingual framework meets Portuguese grammarians

2.1 The core-oblique dichotomy in the Universal Dependencies framework

In the UD annotation guidelines, the argument/adjunct issue follows the same direction since the first version of the project: in view of stated difficulties which are present in a good number of languages that make up the project, UD decided to eliminate the distinction between argument and adjunct in favor of the core-oblique dichotomy.

[1] Transl. "The players are divided **into the ten rooms** of the accommodation, equipped with minibar, air conditioning, television and telephone."
[2] Transl. "Pope appoints woman **to secretary**"
[3] Transl. "PDT aims to reduce federal taxes **to four**."

Marneffe et al. [4] explain that "the core-oblique distinction has to do with the morphosyntactic encoding of dependents, not with their status as obligatory or selected by the predicate" (p. 268). Starting from the idea that some dependency relations are more equally encoded than others across languages, the core terms are those that would be less variably encoded and occur in the same way on the surface, being the subject and the bare object – when it occurs in an "unmarked" way. The criteria for defining what are marked or unmarked forms of the subject and object, as noted by Marneffe et al., are specific to each language, however, some criteria are recurrent, among which we highlight:

i Verbs usually only agree with core arguments.
ii Core arguments often appear as bare nominals while obliques are marked by adpositions or other grammatical markers.
iii Valency-changing operations such as passive, causative, and applicative are often restricted to the promotion or demotion of core arguments.

Considering the criteria, we conclude that phrases preceded by a preposition (item [ii]), when valency-changing operations are not allowed (item [iii]), cannot be core terms.

Zeman [11] notes that a simple criterion for distinguishing between core and oblique in the English treebank is the presence or absence of a preposition, a posture that could also be adopted for the Portuguese language. Thus, a verbal argument is *obj* (direct object) when it is not preceded by a preposition, it is *iobj* (indirect object) only when there is already a direct object in the sentence and this indirect object must necessarily be an oblique pronoun, as it occurs in the dative case and can be un-prepositioned, and *obl* for all other cases, both of prepositional arguments and of adverbial adjuncts.

For treebanks which previously differentiated both classes, Zeman [11] proposes a subspecification from the oblique, the *obl:arg* relation, to be used when, in addition to being prepositional, the phrase is also considered an argument of the verb. Thus, the tags change labels, but the difficulty of distinguishing the argument from the adjunct remains – the lack of consensus, in the grammatical tradition, between indirect object and adverbial adjunct, appears in the Portuguese UD between *obl* (verb-dependent, prepositional) and *obl:arg* (also verb-dependent and prepositional), the first being an adverbial adjunct and the second a verbal argument, traditionally named *indirect object*.

2.2 The argument-adjunct distinction in Portuguese grammatical literature

We consulted different Portuguese grammars about the phenomenon of prepositional phrases attached to verbs.

An essential element for Vilela and Koch [10] in the argument-adjunct distinction is the interrogation directed to the verb in order to identify those terms that "are installed in the very meaning of the predicate" (transl., p. 347). If the term answers the questions "who, which, what, where, how much, how" asked

to the verb, it is an argument; if, on the other hand, the phrase answers the questions "where, why, how, when", it is an adverbial adjunct. We see, however, that there are questions that are repeated in the two classifications (where, how and when), which are thus useless questions for distinguishing between the classes. In the sentences below, where in both (a) and (b) "Francisco" answers the question "*quem* colocou/descobriu" ("*who* put/discovered it"), "Francisco" is classified as an argument (of the subject type), but there is difficulty in classifying the phrase "na prateleira", as in both sentences the phrase answers the questions "*onde* colocou/descobriu" ("*where* they put/discovered it"), an answer that fits both the argument and the adjunct classifications, according to the authors criteria.

a O Francisco colocou a enciclopédia **na prateleira**. (transl. "Francisco put the encyclopedia **on the shelf**.")
b O Francisco descobriu a enciclopédia **na prateleira**. (transl. "Francisco discovered the encyclopedia **on the shelf**.")

In this case, the authors' "intuitions" (VILELA & KOCH [10], p. 348) would tell them that, for the verb "to put", "on the shelf" is an argument, while for the verb "to discover" it is an adjunct. The sense of intuition understood by the authors of the grammar is similar to that criticized by Borges Neto [5] in a similar context, when he provokes: "Perhaps illiterates may have ´'intuitions" about the language, linguists recall analyzes with whom they had contact" (transl., p. 69). The author suggests that this "intuition" is just a process of reaffirming the same categories by repeating analyzes already carried out by the grammatical tradition.

Vilela and Koch look for "supplementary criteria" to justify their intuition. They consider that by deleting an adjunct, the sentence would remain complete – according to them, one can say "Francisco discovered the encyclopedia Ø" and the sentence remains complete, but it would not be acceptable to end the other sentence in "Francisco put the encyclopedia Ø" without the place complement.

We carried out a brief exploration to verify the claim that the verb "to put" requires a place complement. We queried the corpus "todos juntos", in the AC/DC service of Linguateca[4], and it returned 313,047 occurrences of the verb "colocar" ("to put"). At the beginning of the list, we find a small number of sentences using the verb without the prototypical place complement, discrediting the authors' "intuition":

1. Para aproveitar o contra-ataque, Ramirez vai **colocar** os volantes Ney e Cristóvão exercendo uma forte marcação no meio-campo .[5]
2. Para situar nosso questionamento no modelo lógico da Política Nacional de Monitoramento e Avaliação da Atenção Básica8, é necessário **colocar** a

[4] Available at: https://linguateca.pt/ACDC. Accessed on 11 Jan. 2022.
[5] Transl. "To take advantage of the counterattack, Ramirez will **put** the midfielders Ney and Cristóvão exerting a strong marking in the midfield."

aquisição de novos conhecimentos e a melhoria do desempenho do Sistema Único de Saúde (SUS) como suas principais finalidades.[6]

Bechara [3] is careful not to call those prepositional phrases as neither prepositional objects (which is the case of "Amar **a Deus** sobre todas as coisas"/ lit. "Love **to God** over all things."), nor indirect objects ("The director wrote letters **to parents**"). He names them "relative complement", being similar to the direct object in semantic-syntactic properties, except for the presence of a preposition.

Bechara indicates that each verb is accompanied by its own preposition by what he calls "grammatical servitude". Thus, "depender de" ("to depend on"), "competir com" ("compete with") and "agregar a" ("aggregate to") are predictable, although there are exceptions: first, the case in which the norm allows the use of more than one preposition ("ela se parece ao/com o pai"/ "she resembles to/with her father"), and second, the case of linguistic variation (diatopic, diastratic and diaphasic), as with the verbs "socorrer", "contentar" and others, that can be used with or without a preposition. This position is updated by Bagno [2], who presents examples of historical change, and not just variations of Brazilian Portuguese, as in the cases of "desagradar (a) alguém" (lit. "displease (to) someone"), "desobedecer (a) algo" (lit. "disobey (to) something"), "aspirar (a) algo" (lit. "aspire (to) something"), etc.

Finally, Bechara reminds that not all scholars agree that relative complements should be considered arguments: "Taking into account exclusively the semantic aspect, many prefer to consider such terms as circumstantial or adverbial adjuncts (...)" (BECHARA [3], p. 446). As we will see in section 3, our proposal to annotate the verbal arguments and adjuncts is endorsed by this position.

3 Methodology

Our annotation of the *obl* and *obl:arg* relations is motivated by the need to both achieve internal consistency and to make the analyzes informative, distinguishing sentences (1) from (2), which will be *obl:arg* and *obl* (argument and adjunct, respectively), and equating (3) and (4), which will be *obj* and *obl:arg* (both arguments).

1. Gostar **de sorvete**. (lit. "To like **to icecream**.")
2. Viajou **de carro**. (transl. "Traveled **by car**.")
3. Assistiu **o filme**. (transl. "Watched **the movie**.")
4. Assistiu **ao filme**. (lit. "Watched **to the movie**")[7]

[6] Transl. "To place our questioning in the logical model of the National Policy for Monitoring and Evaluation of Primary Care8, it is necessary to **put** the acquisition of new knowledge and the improvement of the performance of the Unified Health System (SUS) as its main purposes."

[7] As noted by a reviewer, we could consider the preposition a pleonastic element, as the sentence admits passive alternation. However, this criterion is not absolute. The verb "gostar de" (lit. "to like to") can admit the passive alternation in informal

Our strategy is to look at the meaning of the prepositional phrase in the sentence – if its meaning is the meaning traditionally associated with an adverb (time, place, manner, finality, causality, conformity), we annotate it as *obl* and, in the absence of adverbial semantics, it is an *obl:arg*. Thus, we shift the syntactic focus on the demand made by the verb to semantic-contextual features of the noun phrase associated with it.

The corpus is composed of 20,210 verb occurrences (1,080 different lemmas), being very expensive to analyze them case by case. We bootstrapped the annotation from Stanza [8] and transformed it to our proposal using the established semantic-discursive criterion. Our strategy was conducted three steps:

718 verbs are associated with the preposition "em" (lit. "in")
371 verbs are associated with the preposition "com" (lit. "with")
307 verbs are associated with the preposition "a" (lit. "to")
305 verbs are associated with the preposition "para" (lit. "for")
250 verbs are associated with the preposition "de" (lit. "of")

Table 1. 5 first prepositions that are most associated to verbs in PetroGold

i In a spreadsheet, we list all the verbs that, indirectly, are associated with prepositions (in the dependency model, the preposition depends on a noun, which is dependent on the verb). We organized the spreadsheet by preposition, and a sample of the five most popular prepositions among verbs can be viewed in the Table 1. Four annotators were responsible for evaluating whether, for each combination of verb + preposition, the prepositional phrase could be an argument of the verb. The only focus of this step is to separate verbs that can have an argument from those that never do, because while any predicate can have an adverbial adjunct, not all can have an argument. When the annotators could not think of an argument for the verb + preposition combination, they looked at the occurrences in the corpus to make sure there was not one. This step is intermediate, and its goal is to facilitate the corpus review process, in order to minimize the number of occurrences that will be reviewed.

ii Automatic changes are performed in the corpus using the data from the spreadsheet reviewed by the annotators. Thus, if a combination of verb + preposition, such as "acarretar em" ("result in"), appeared in the spreadsheet as possibly having an argument, all occurrences of "acarretar em"

register, although being typically an indirect transitive verb, as well as possibly any other verb. This way, it is reasonable to equate sentences (1) and (4), since both have prepositional phrases which are arguments of the verb (*obl:arg*), regardless of passive alternation possibility: "A nova empreitada do Linkedin permitirá que os produtores de conteúdo vejam quantas vezes um texto **foi gostado**, comentado e compartilhado."

("result in") became an argument (*obl:arg*), regardless of whether they are correct, like the underlined words in the following sentence:

* Segundo Souza (2009), a estabilidade conferida às emulsões devido à presença dos agentes emulsionantes naturais **acarreta**, **em geral**, **em um incremento** significativo na sua viscosidade[8]

iii We contrast the automated changes made in step (ii) with the original parsing. At this point, each annotator is guided to cases where spreadsheet and parser diverged. For example, in the previous sentence, the parser tagged "geral" ("general") as an adverbial adjunct, but the spreadsheet signaled "acarretar em" ("result in") as a verb with an argument. Annotators, aid by a specific tool to contrast two analyses[9], should just select the correct one.

The goal of the strategy was to reduce the time needed to correct the arguments and adverbial adjuncts with prepositioned phrases, as we only verified the occurrences in which there was a discrepancy between the spreadsheet annotation and the parser annotation.

We provide the spreadsheet[10] which we used to indicate what verbs, when related to which prepositions, can have the (prepositioned) noun as their complement. The spreadsheet includes all verb lemmas that relate to prepositions in the corpus, with a noun example for each entry. It should be noted that our objective with the spreadsheet is not to provide the community with any kind of definitive list of the transitivity of verbs, since it played a small part of a bigger strategy to correct the annotation of difficult sentences. However, from the point of view of linguistic description, it may be interesting to obtain a list of verbs and prepositions, and it is still possible to rearrange it, in alphabetical order, obtaining a list of all the prepositions that relate to each of the verbs in the corpus instead of distributing verbs by preposition as we have provided.

4 Results

We have made available the modifications related to arguments and verbal adjuncts in version 2 of PetroGold [9]. As a result, 1,488 tokens were modified in the corpus, which corresponds to 14.8% of sentences being modified.

In Figure 1 we present the distribution of lemmas by the frequency they occur with an argument. In the figure, we also classify as arguments the object clauses, annotated as *xcomp* and *ccomp* in UD, a position also defended by Przepiórkowski and Patejuk [7].

We removed from the analysis all verbs in the participle form and verbs with the expletive pronoun "se" dependent on them. In sentences with participles,

[8] Transl. "According to Souza (2009), the stability conferred on emulsions due to the presence of natural emulsifying agents **results**, **in general**, **in a significant increase** in their viscosity"

[9] Available at: https://github.com/alvelvis/conllu-merge-resolver. Accessed 21 Feb. 2022.

[10] Available at https://petroles.puc-rio.ai, along with PetroGold v2.

it is difficult to automatically distinguish which verbs do not accept complement ("[isso] ocorre Ø"/"[this] occurs Ø", sentence (1)) and which could accept it ("[alguma reação] hidrolisou [a poliacrilamida]"/"[some reaction] hydrolysed [a polyacrylamide]", sentence (2)). In sentences with "se", there is also doubt about the presence or not of an object: sometimes the verb actually works as an intransitive ("[algo] se sobressai Ø"/"[something] stands out Ø", sentence (3)), sometimes the verb could be interpreted as accepting a complement ("[algum fenômeno natural] assentou [as rochas]"/"[some natural phenomenon] based [the rocks]", sentence (4)). As we still do not have a systematic study of these cases, we prefer to leave them aside for the moment.

1. Isso pode ter **ocorrido** devido o clorofórmio extrair também o tensoativo.[11]
2. Viscosidade vs. taxa de cisalhamento de poliacrilamida **hidrolisada**.[12]
3. Estas fontes **se sobressaem** no mapa de amplitude do sinal analítico referido acima.[13]
4. As rochas da Bacia Sanfranciscana **assentam-se**, em discordância erosiva e angular, sobre rochas paleoproterozóicas do embasamento.[14]

As a result of the elimination of these types of verbs from the analysis, the study counted with 9,653 verbal occurrences that are distributed in 719 lemmas, 66% of the total verbal lemmas in the corpus. We can see how many verb lemmas are never accompanied by object, how many are accompanied by objects less than 30% of the time, between 30% and 70%, more than 70% of times, and how many are always accompanied by objects (*obj*, *iobj*, *obl:arg*, *xcomp* and *ccomp*).

The vast majority of verbs in PetroGold are always followed by an argument. Secondly, we have verbs that most often have an argument, thirdly we have those that never have an argument, then those that are exactly in the middle, not trending towards neither transitivity nor intransitivity, and finally, those that almost never have an argument.

This slice of lemmas that are in the middle, between "never" and "always", corresponds to 25.8% of the lemmas in the corpus. That is, a quarter of the verbal lemmas are exactly halfway between intransitivity and transitivity. For all these cases, it cannot be said, on the one hand, that when they lack a complement the sentence is incomplete, and, on the other hand, it cannot be said that the verb does not allow a complement without making a considerable mistake with that statement. This type of statistical information that we obtained escapes a categorical description of verbal subcategorization, and it is only possible because we have annotated the adjunct-argument distinction in a way that avoided transitivity as an intrinsic property of verbs.

[11] Transl. "This could have **happened** due to the chloroform extracting the surfactant as well."

[12] Tranl. "Viscosity vs. shear rate of **hydrolyzed** polyacrylamide."

[13] Transl. "These sources **stand out** in the analytical signal amplitude map referred to above."

[14] Transl. "The rocks of the Sanfranciscana Basin are **based**, in erosive and angular unconformities, on Paleoproterozoic rocks of the basement."

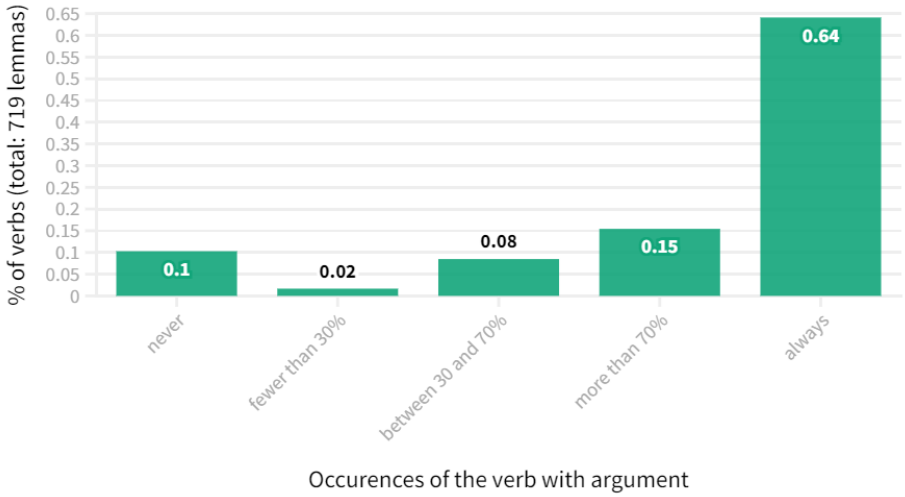

Fig. 1. Distribution of verbal lemmas in PetroGold by the frequency in which they occur with or without argument

5 Concluding remarks

This paper takes up the subject of verbal arguments and adjuncts with an empirical approach. First, we studied the status of the indirect objects and adverbial adjuncts in the Universal Dependencies guidelines, where we have seen enough arguments disfavoring this kind of distinction, while still leaving space for each treebank to discuss if and how they will annotate particular cases. Portuguese grammars brought many different criteria to establish the boundaries between both classes, but we saw they are insufficient when confronted with real language data. Then, we proposed a semantic-dicursive criterion, presented our annotation methodology and showed the results, which affected 14.8% of sentences in PetroGold and are featured in its second version.

References

1. Afonso, S., Bick, E., Haber, R., Santos, D.: Floresta sintá(c)tica: a treebank for Portuguese. In: Rodrigues, M.G., Araujo, C.P.S. (eds.) Proceedings of the Third International Conference on Language Resources and Evaluation (LREC 2002). pp. 1698–1703. ELRA, Paris (29-31 de Maio 2002), http://www.linguateca.pt/documentos/AfonsoetalLREC2002.pdf
2. Bagno, M.: Gramática pedagógica do português brasileiro. Parábola Ed. (2012)
3. Bechara, E.: Moderna gramática portuguesa. Nova Fronteira (2012)
4. De Marneffe, M.C., Manning, C.D., Nivre, J., Zeman, D.: Universal dependencies. Computational linguistics **47**(2), 255–308 (2021)

5. Neto, J.B.: Morfologia: conceitos e métodos. Colóquios linguísticos e literários: enfoques epistemológicos, metodológicos e descritivos. Teresina: Edufpi pp. 53–72 (2011)
6. Nivre, J., De Marneffe, M.C., Ginter, F., Goldberg, Y., Hajic, J., Manning, C.D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., et al.: Universal dependencies v1: A multilingual treebank collection. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). pp. 1659–1666 (2016)
7. Przepiórkowski, A., Patejuk, A.: Arguments and adjuncts in universal dependencies. In: Proceedings of the 27th International Conference on Computational Linguistics. pp. 3837–3852 (2018)
8. Qi, P., Zhang, Y., Zhang, Y., Bolton, J., Manning, C.D.: Stanza: A Python natural language processing toolkit for many human languages. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations (2020), https://nlp.stanford.edu/pubs/qi2020stanza.pdf
9. de Souza, E., Freitas, C.: Polishing the gold – how much revision do we need in treebanks? In: Proceedings of the I Universal Dependencies Brazilian Festival (UDFest-BR) (2022)
10. Vilela, Mário; Koch, I.V.: Gramática da língua portuguesa: Gramática da Palavra, Gramática da Frase, Gramática do Texto/Discurso. Almedina (2001)
11. Zeman, D.: Core arguments in universal dependencies. In: Proceedings of the fourth international conference on dependency linguistics (DepLing 2017). pp. 287–296 (2017)

UDConcord: A Concordancer for Universal Dependencies Treebanks

Lucas Gabriel Mendes Miranda and Thiago Alexandre Salgueiro Pardo

Interinstitutional Center for Computational Linguistics (NILC)
Institute of Mathematical and Computer Sciences, University of São Paulo
São Carlos – SP, Brazil
lucasgmm@usp.br, taspardo@icmc.usp.br

Abstract. This paper presents UDConcord, a concordancer web application. The tool is designed to be visual and easy-to-use, working with treebanks that were annotated using the Universal Dependencies international model. It allows users to upload a treebank in the CoNLL-U format. After the upload, users can search for terms and linguistic categories of interest in the treebank. Because the tool is a concordancer, the search results are composed of sentences with occurrences of the searched elements displayed in a concordance list. That means that each sentence with a matching term will be displayed in a row, with the found term centralized and highlighted, accompanied with other information selected by the user to be visualized. UDConcord also allows users to easily modify sentence annotation. Finally, UDConcord makes it possible for users to download the treebank's updated version, with every change made up until that point.

Keywords: Concordancer · Universal Dependencies · Treebank.

1 Introduction

The Universal Dependencies project (UD) [6, 5] is an initiative that seeks to standardize dependency-based treebank annotation. The project establishes a set of guidelines that must be followed during the construction of these types of treebanks.

One of the UD guidelines states that a treebank must be represented in CoNLL-U file format. Such files must contain all the sentences in the treebank, along with its annotation data, which is token/word-based. According to the guidelines, every word/token in a sentence must have a series of values distributed over 10 fields (columns), including the word's part-of-speech tag, lemma, dependency relation, and other properties.

Those guidelines introduce a series of benefits to treebank annotation and design, but, to people who are manipulating the CoNLL-U files, they also can be hard to deal with. Particularly, problems might arise when users are trying to query a treebank. For example: if a user wanted to search for two consecutive words with specific part-of-speech tags in a treebank, he/she would not be able

to do it easily, thanks to the fact that treebanks are formatted in a table-like manner and that the available search tools are not straightforward to use or to install.

To help in this front, we present UDConcord, which is a concordancer web application to simplify search, analysis, and edition of CoNLL-U files. UDConcord seeks to be easy to use and simple (without the need to install the tool or to learn some kind of search syntax), allowing users to query treebanks, presenting the query results in a form a concordance. It also allows users to edit sentences' annotation data easily.

2 Related Work

There are a number of applications that allow users to query treebanks. Unfortunately, most of them have at least one of the following two issues:

- They do not allow users to edit the treebanks. For example, TüNDRA [4] and Grew-match [2] are both tools for *only* searching treebanks;
- They have user interfaces that are relatively complex, which makes it harder for inexperienced users to use them. For example, ConlluEditor [3] is one of the tools that offer both searching and editing of treebanks. However, its user interface suffers from an excess of buttons and options, which can be confusing for new users. Other interesting example to cite is Arborator-Grew [1], which, besides intended to be more user friendly and to include more functionalities (allowing to search, edit and visualize trees), requires the user to master some search syntax and to use an interface with too much information.

These two problems were considered when designing UDConcord, especially the second one. Our main goal was to create a tool that is simple, intuitive, and easy to use, in order to make it friendlier to inexperienced users.

Below, we highlight two tools that are similar to ours and that somehow inspired us to propose our tool, pointing UDConcord's strengths in comparison.

Interrogatório [8] is an environment for searching and editing universal dependency-based treebanks. It is written with Python and Javascript. However, the app is supposed to only run locally and to be accessible with a web browser through localhost. That could be a problem, because, in order for the user to use Interrogatório, he/she must install it first, which can be confusing if they are less experienced. UDConcord does not suffer from this, because it is a web application accessible through the web.

To make queries with Interrogatório, the user must learn a query language that is similar to Python code. Again, if the user is unfamiliar with Python, he/she might experience some confusion in the process of learning the language. To avoid this possibility during the usage of UDConcord, we implemented the query feature with the use of a form. That is, to make queries with UDConcord, users do not need to learn a query language, they only need to fill a form with their query's parameters and click on the search button.

Another tool very similar to UDConcord is Grew-match [2]. Like Interrogatório, it is written in Python and Javascript and allows users to query treebanks using a query language, which we already commented that can be confusing for less experienced users.

Like UDConcord, Grew-match is a web application accessible through the web. However, Grew-match's users can only query a set of predefined UD-based treebanks, which is fairly large, while in UDConcord's users have to upload their treebanks in order to query them. We believe that this is useful because it offers more flexibility to users.

Another important distinction between UDConcord and Grew-match is that the latter does not allow its users to edit the CoNLL-U from the queried treebank, while UDConcord does.

3 The Architecture of UDConcord

UDConcord is an application with three main components: a back-end, a front-end, and a reverse proxy. All these three components are installed in different Docker containers, like shown in Figure 1.

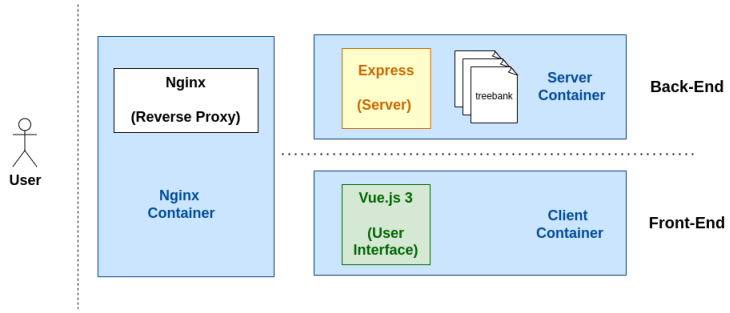

Fig. 1. UDConcord's Architecture.

In the following subsections, we will specify what exactly each component does.

3.1 Back-end

The back-end contains code concerned with the search and storage of treebanks. It is entirely built with Javascript, with the *Node.js* runtime.

When the user uploads a treebank to our application, the back-end is responsible for parsing it to an array of Javascript objects (with each object being a sentence from the treebank) and saving it as a JSON file.

When the user searches a treebank, the back-end reads the JSON file that corresponds to the treebank and looks for sentences that satisfy the user's search

conditions. Once they are found, they are sent back to the user. This approach guarantees queries with reasonable wait times for CoNLL-U files with at least 20 MB of size.

All these functionalities are exposed to the front-end through an API built with the *Node.js* framework *Express*.

3.2 Front-end

The front-end is composed of code concerned with the user interface. More specifically, it contains all the HTML, CSS and Javascript used to build it. It is important to note, that, for the Javascript language, we used the *Vue.js 3* framework, which immensely facilitated our work. Furthermore, certain components, like text inputs, tables, and buttons were taken from the *PrimeVue* component library. The front-end communicates to the back-end through HTTP requests to the API in the latter.

3.3 Reverse Proxy

The reverse proxy is a server responsible to direct users' requests to another appropriate server. For example, when the user enters UDConcord's web address, he will request the server holding the static files (HTML, CSS, and Javascript files) to download them. The reverse proxy will receive this user's request and direct it to the server running in the client container. Similarly, when the user makes a request to the back-end API, the reverse proxy will direct it to the Express server, which will process the request.

Our reverse proxy is configured using *NGINX*, which is a software commonly used for this purpose.

4 The Concordancer

UDConcord has four main features:

- The possibility of searching for terms and linguistic categories of interest in an uploaded treebank;
- To display every sentence with an occurrence of the searched elements;
- The possibility of editing the corresponding CoNLL-U files;
- Enabling the user to download the updated version of the treebank (with all the made changes);

All of these features are further described in the following subsections.

4.1 Searching/querying

UDConcord's main feature is to let users query a treebank. However, to do that, they must first upload the treebank to the system. The upload screen is available on the home page of the app. It is the first thing users see when they enter the website.

To upload the treebank, users just have to click on the blue button labeled *Choose file* and select in their operating system the CoNLL-U file that represents the treebank.

Simple Searches After uploading the treebank, users will be redirected to the search screen. In that screen, he/she will have to specify their query parameters filling out a search input, which is shown in Figure 2.

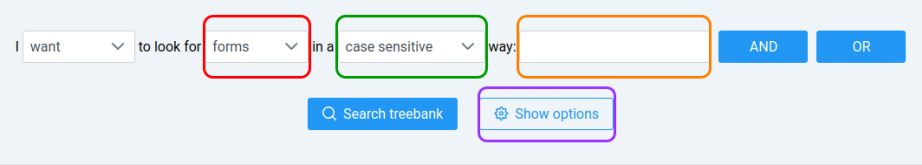

Fig. 2. Search input.

The selector highlighted in red allows the user to choose between five properties (all defined in the UD specifications) to search for in the uploaded treebank: forms, lemmas, part-of-speech tags (POS tags), dependency relations (deprels) and features (feats).

In addition, the selector highlighted in green allows the user to choose if he/she wants their search to be made in a case-sensitive or insensitive way. On the other hand, the input field highlighted in orange is the one where the user must enter the values that should be searched in the treebank.

The button highlighted in purple shows some options about how the search results should be displayed. For example, it allows the user to indicate whether the part-of-speech tag of each token in every sentence should be displayed on the results page or not.

Note that the whole search input is organized in a way that it forms a sentence (from left to right). This was done this way to improve usability and to decrease the user's learning curve while he/she is learning how to use the tool.

Complex Searches Users can also build complex search patterns with the use of the logical conditions AND, OR, and NOT. The logical AND and the logical OR are represented by new rows in the search input, while the logical NOT is represented by a value in the selectors highlighted in black in Figure 3.

The AND and OR logical conditions can be added to the search by clicking on any of the buttons labeled *AND* and *OR* highlighted in pink in Figure 3. The

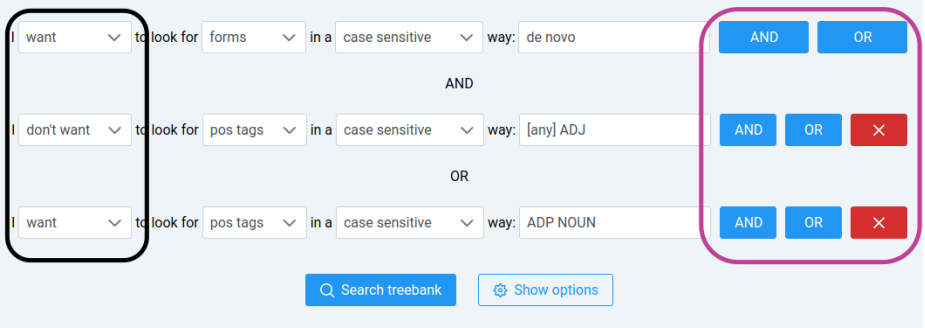

Fig. 3. Search input with logical conditions.

logical condition is added below the row that contains the clicked button. To delete a logical AND or logical OR, the user must click on the red button that is on the same row that he/she wants to remove.

Like we mentioned before, the logical NOT is represented by a value in the selectors highlighted in black in Figure 3. This selector has the following two possible values:

- "want": the row will be evaluated *without* a logical NOT;
- "don't want": the row will be evaluated *with* a logical NOT.

To better clarify this manner of searching, we will describe a short example below.

Example: searching with logical conditions Suppose a user wants to search for the following 2-gram in a treebank:

- The first token of the 2-gram has the form "de";
- The second token of the 2-gram has the form "novo" *AND* its part-of-speech tag is not "ADJ". Alternatively, the token can have a part-of-speech tag of "NOUN".

To make this search using UDConcord, users have to fill out their search input like shown in Figure 3. Three rows of input are necessary. The first is the initial one and the other two are for the *AND* and *OR* logical conditions.

In the first row, we select the "forms" option in the second selector and enter "de novo" in the text input. That is because we want to search for two tokens, in that order, with that specific forms. Furthermore, we select the "want" option in the first selector because we want the found tokens to have their forms equal to the ones specified in the text input.

In the second row, which defines the logical AND, we select "pos tags" in the second selector. In the first selector, we choose "don't want" because we need the found tokens to have their part-of-speech tags different than the ones

specified in the input text. Then, in that input text, we enter "[any] ADJ". The "[any]" is a special word in our program that signalizes that a token can have any value. In the case of our example, the first token can have any part-of-speech tag value. Note that the "[any]" is not affected by the "don't want". In addition, the "[any]" keyword can also be used to search for skip-grams, by putting it in the place of the token that should be skipped in the query.

Finally, in the third row, which defines the logical OR, we select "pos tag" in the second selector. Furthermore, in the first selector, we choose "want" because we need the found tokens to have their part-of-speech tags equal to the ones specified in the input text. In that input text, we enter "ADP NOUN". Then, we click on the "Search treebank" to start the search.

Note that there is a specific order of precedence in the evaluation of the logical conditions. First, NOT logical conditions are evaluated, then AND, then OR.

4.2 Displaying the Results

After defining their search parameters and clicking on the "Search treebank" button, the search will be made, and the results are going to be displayed on the screen in the form of a concordance. That means that each sentence with a matching n-gram will be displayed in a row, with the found n-gram centralized and highlighted in blue.

Visualization Options: Displaying Part-of-speech Tags As mentioned in subsection 4.1, the button highlighted in purple (Figure 2) shows some options about how the search results should be displayed. Clicking on it exposes three checkboxes with the following options: *POS Tags*, *Dependency relations*, and *Features*.

These options enable the user to choose a few extra data he/she might want to be displayed on the results page. For example, if he/she checked the box "POS tags", the part-of-speech tags of every token would be displayed in the search results after a "/" character.

Visualization Options: Displaying Dependency Relations Note that users can also choose to display dependency relations on the results. In that case, these dependency relations would be displayed as shown in Figure 4.

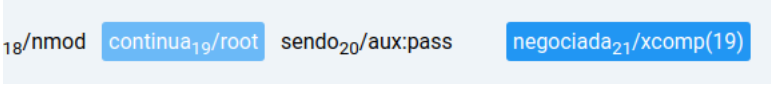

Fig. 4. Display of dependency relations in the results.

The darker blue highlights the matched token/n-gram in the search and the dependent in the dependency relation. The number between parentheses is the token's head's id. Before the left parenthesis, there is also the label of the dependency relation. The lighter blue highlights the head of the dependency relation. So, Figure 4 is indicating that there is a dependency relation of type "xcomp" between "negociada" (dependent) and "continua" (head). Also, note that each token has its id subscripted next to it.

4.3 Editing Sentence Annotation

If the user double clicks on a sentence in the results, he/she will be redirected to the corresponding CoNLL-U. There, they will be able to edit the sentence's annotation data.

The editor is organized in a table-like manner, with each line of the sentence's CoNLL-U corresponding to one row in the editor. Therefore, a row can have the sentence's metadata, or token/word annotation – in that case, the row is divided into 10 cells: one for each field defined in the CoNLL-U format. Editing a table cell is the same as editing one value in the CoNLL-U annotation. Currently, UDConcord does not validate the values entered in the cells.

If the user right-clicks on a row, UDConcord will present the following three options:

- Add a row below the one clicked;
- Add a row above the one clicked;
- Delete the clicked row.

It is important to note that if the user removes or adds a row, the ids and the head field values from each row are automatically adjusted.

When the user finished the editing on the CoNLL-U file, he/she just have to press the button labeled "Save changes" (at the bottom right) to save their changes. Then, he/she can go back to the results page by clicking on the button labeled "Go back" (also at the bottom right).

4.4 Downloading the Edited Corpus

UDConcord offers several options to download an uploaded treebank after the user made changes to it. These options can be found after clicking on the arrow inside the blue button labeled "Download" located at the bottom right of the results page:

- "Download treebank (.conllu)": clicking on it starts the download of the uploaded corpus, with every change made by the user;
- "Download search results (.conllu)": clicking on it starts the download of the CoNLL-U of every sentence returned by the search. This is great if you want to filter the CoNLL-U of only a few sentences;

– "Download search results (.csv)": clicking on it starts the download of the search results in a .csv format (not the CoNLL-U, just the sentences themselves);
– "Download search results (.txt)": clicking on it starts the download of the search results in a .txt format (not the CoNLL-U, just the sentences themselves).

5 Caveats

As we previously mentioned, one of UDConcord's main goals is to provide an easy-to-use experience to its users. In order to do that, we implemented a query interface that is simple and intuitive. However, this simplicity come with limitations on the kind of queries that can be made with the tool. For example, currently, there is no way to search for head-dependent relations, like a NOUN that is a dependent of the verb "eat" with the "obj" relation. Our tool does not support this kind of query, because its complexity did not allow it to be well translated to our query interface in a simple way. For users that want to make use of a more robust query mechanism that is not covered by UDConcord, we recommend tools like Grew-match and Interrogatório, which allow them to search using standard query languages.

6 Final Remarks

UDConcord was designed to be simple and easy to use. We developed it to ease the task of working with Universal Dependencies-based treebanks. This is especially useful to non-computer savvy users, who might encounter some problems when dealing with the CoNLL-U files.

While simplicity is the tool's main characteristic, we also focused on providing the necessary features for quality treebank analysis, search, and annotation.

UDConcord is available at https://udconcord.icmc.usp.br/. Other related resources and tools may be found at the web portal[1] of the POeTiSA project (which stands for "POrtuguese processing - Towards Syntactic Analysis and parsing"). In this project, UDConcord has been used as support to the construction of the Porttinari treebank for Brazilian Portuguese [7].

Acknowledgements

The authors are grateful to the Center for Artificial Intelligence of the University of São Paulo (C4AI[2]), sponsored by IBM and FAPESP (grant #2019/07665-4).

[1] https://sites.google.com/icmc.usp.br/poetisa
[2] http://c4ai.inova.usp.br/

References

1. Guibon, G., Courtin, M., Gerdes, K., Guillaume, B.: When collaborative treebank curation meets graph grammars. In: Proceedings of The 12th Language Resources and Evaluation Conference. pp. 5293–5302. European Language Resources Association, Marseille, France (May 2020), https://www.aclweb.org/anthology/2020.lrec-1.651
2. Guillaume, B.: Graph matching and graph rewriting: GREW tools for corpus exploration, maintenance and conversion. In: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations. pp. 168–175. Association for Computational Linguistics, Online (Apr 2021). https://doi.org/10.18653/v1/2021.eacl-demos.21, https://aclanthology.org/2021.eacl-demos.21
3. Heinecke, J.: ConlluEditor: a fully graphical editor for Universal dependencies treebank files. In: Universal Dependencies Workshop 2019. Paris (2019), https://github.com/Orange-OpenSource/conllueditor/
4. Martens, S.: Tündra: A web application for treebank search and visualization. In: Proceedings of The Twelfth Workshop on Treebanks and Linguistic Theories (TLT12). pp. 133–144. Association for Computational Linguistics (2013), http://bultreebank.org/TLT12/TLT12Proceedings.pdf
5. Nivre, J.: Towards a universal grammar for natural language processing. In: Proceedings of the 16th International Conference on Intelligent Text Processing and Computational Linguistics. pp. 3–16. Springer, Cham, Switzerland (2015)
6. Nivre, J., de Marneffe, M.C., Ginter, F., Hajič, J., Manning, C.D., Pyysalo, S., Schuster, S., Tyers, F., Zeman, D.: Universal dependencies v2: An evergrowing multilingual treebank collection. In: Proceedings of the 12nd International Conference on Language Resources and Evaluation. pp. 4034–4043. European Language Resources Association, Marseille, France (2020)
7. Pardo, T., Duran, M., Lopes, L., Felippo, A., Roman, N., Nunes, M.: Porttinari - a large multi-genre treebank for brazilian portuguese. In: Proceedings of the XIV Symposium in Information and Human Language (STIL). pp. 1–10. Sociedade Brasileira de Computação, Porto Alegre, RS, Brasil (2021). https://doi.org/10.5753/stil.2021.17778, https://sol.sbc.org.br/index.php/stil/article/view/17778
8. de Souza, E., Freitas, C.: ET: A workstation for querying, editing and evaluating annotated corpora. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 35–41. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic (Nov 2021), https://aclanthology.org/2021.emnlp-demo.5