Parsing Tools for Italian Phraseological Units

Alfredo Milani[1], Valentina Franzoni[1(✉)], and Giulio Biondi[2(✉)]

[1] University of Perugia, Department of Mathematics and Computer Science, 06123 Perugia, Italy
alfredo.milani@ieee.org, valentina.franzoni@dmi.unipg.it
[2] University of Florence, Department of Mathematics and Computer Science, 50134 Florence, Italy
giulio.biondi@unifi.it

Abstract. Phraseological complexity plays a critical role in assessing the language competence level necessary to understand or produce text by a language learner and automated tools supporting international certifications for second languages. Appropriate syntactic-based parsing tools for pos/tagging texts are required to efficiently and correctly identify the components of phraseological units to apply computational and corpus-based measures. In this paper, we consider some of the main syntactic parsing tools publicly available for the Italian language and compare them in the framework of a project aiming at characterizing the phraseological complexity of a corpus produced by learners of Italian as a second language.

Keywords: Phraseological complexity · Corpus-based measures · PoS tagging · Parsing tools

1 Introduction

A single, widely accepted definition of phraseological complexity for second language learning research is not yet available in the literature. Attempts [2,15] to define the theoretical aspects of complexity and its measurement draw on the definition given by Rescher in [20]: *a matter of the number and variety of an item's constituent elements and of the elaborateness of their interrelational structure' and share the view that it can be evaluated across various language domains (including syntax, lexis, and morphology).* Subsequent studies [17,18] emphasized the importance of phraseological complexity in assessing the level of language proficiency required to understand or produce a text by a second language learner, particularly at the intermediate/advanced level. The same issue applies to automated text analysis systems for international second language (L2) certifications. There is therefore a call for a definition of appropriate measures of phraseological complexity because existing measures for the quantification of the linguistic complexity of a text are mostly based on lexical diversity

© Springer Nature Switzerland AG 2021
O. Gervasi et al. (Eds.): ICCSA 2021, LNCS 12955, pp. 427–435, 2021.
https://doi.org/10.1007/978-3-030-87007-2_30

[8], e.g., Advanced Guiraud [5] and MTLD [11], and syntactic complexity, e.g., Mean Length of Utterance (MLU) [1] and the amount of subordination per multiclausal unit [19]. Such measures, however, are not capable of recognizing the complex relations between words that frequently co-occur in combinations [6,7], often modifying their meaning and specialization. In [18], the use of Pointwise Mutual Information (PMI) [3] is proposed. The availability of appropriate Natural Language Processing (NLP) parsing software to efficiently and correctly identify components of phraseological units, e.g., by performing part-of-speech tagging, is paramount for any subsequent analysis such as the application of computational and corpus-based measures. Because of the peculiarities of each language, tools for Natural Language Processing must be trained on specific text corpora to identify common patterns, word relationships, and, in general, features that define a language. Most of the NLP research has been conducted on the English language, and the same happens in particular for L2 complexity. Thus, there is much wider availability of resources for English than for other languages; however, the increasing amount of digitalized corpora in other languages is driving the development of language-specific tools. In this paper, we consider some of the main syntactic parsing tools publicly available for the Italian language and proceed to a systematic comparison, in the framework of a project aiming at characterizing the phraseological complexity of a corpus produced by learners of Italian as a second language.

Section 2 explores tools for Natural Language Processing for the Italian language; Sect. 3 proceeds to their comparison; in Sect. 4 conclusions are drawn.

2 NLP Tools for Italian Language

2.1 UDPipe

UDPipe (Universal Dependencies Pipe) [25] is an open-source trainable pipeline for tokenization, tagging, lemmatization, and dependency parsing of CoNLL-U files, developed by Milan Straka at the Institute of Formal and Applied Linguistics, Faculty of Mathematics and Physics of Charles University, Czech Republic. UDPipe is language-agnostic and can be trained given annotated data in CoNLL-U format, a standard format used to annotate data at word/token and sentence level. CoNLL-U is the standard format of the Universal Dependencies project [14], which aims at providing annotated treebanks for several languages to facilitate the development of multilingual tools and cross-language learning. To date, over 200 treebanks in more than 100 languages have been published; for the Italian language, 7 treebanks are available, i.e. *ISDT*, *PUD*, *ParTUT*, *PostWITA*, *TWITTIRO*, *VIT*, *Valico*. Italian-ISDT is the most extensive, featuring 14167 sentences and nearly 280k tokens; comprehensive statistics on the text corpora used to create the different treebanks are available at [9]. UDPipe is available in three versions, at different development stages and using different techniques:

UDPipe 1 [23] includes modules to perform tokenization, tagging, lemmatization, and dependency parsing of CoNLL-U files. It is available in different languages, i.e. C++, Python, Perl, Java, C#, R, and as a web service.

UDPipe 2 [22] is a Python-only prototype, which retains the functionalities of UDPipe 1, except for tokenization. UDPipe 2, unlike the previous version, makes use of Deep Learning techniques; on the one hand, this led to an increase in the computational power requirements, balanced by a great improvement (40%–50%) in morphosyntactic analysis performance. Models are available for various languages, including Italian. UDPipe 2 achieved some remarkable results in competitions, including achieving first place in the *CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies* [24] and the EvaLatin 2020 Evaluation Campaign [21], with a specific model trained for the task.

UDPipe 3 is still under early development and will feature C++, Python, and other languages bindings. TensorFlow is chosen as the Deep Learning backend; along with morphosyntactic analysis tools, which were already available in the previous versions, additional modules, e.g. *Entity Linking* and *Named Entity Resolution*, are planned.

2.2 TINT

The Italian NLP Tool (TINT) [16] is an open-source Java-Based NLP pipeline focused on the Italian language, developed at Fondazione Bruno Kessler (FBK), Trento, Italy. It is based on the popular CoreNLP tool developed by the University of Stanford [10], which itself supports six languages, i.e. Arabic, Chinese, English, French, German, and Spanish. Tint has been used as a pre-processing tool for several NLP tasks, e.g. Hate Speech Detection [4], automatic text difficulty rating [12,13].

2.3 SpaCy

SpaCy is an open-source library for advanced NLP, written in Python, currently in its 3.0 version. It features multi-language support, pre-trained models, and word embeddings, and allows the usage of custom models in PyTorch, Tensorflow, and other Deep Learning frameworks. At the time of writing, three pipelines are provided for Italian language, i.e. *it_core_news_sm*, *it_core_news_md*, and *it_core_news_lg*.

3 Experimental Comparison

In this section, we perform a comparison among the three analyzed tools for Natural Language Processing of Italian as a second language.

3.1 Code and License

All the compared tools for Natural Language Processing are open source and currently actively maintained. The copyright license varies as follows:

- UDPipe is available under the Mozilla Public License (MPL) 2.0
- TINT is available under the GNU General Public License (GPL) 3.0
- SpaCy is available under the MIT License.

3.2 Comparison of General Functionalities

In Table 1 a comparison of the general functionalities available in the considered tools for Natural Language Processing is presented. We can see that TINT and SpaCy offer all the complete set of considered functionalities, while for UDPipe functionalities are distributed in different versions of the software and some tasks are under development for futures release.

Table 1. Functionalities Available in UDPipe, TINT, SpaCy

Functionalities	UDPipe	TINT	SpaCy
tokenisation	YES (v1)	YES	YES
sentence segmentation	YES(v1, v2)	YES	YES
morphological analysis	YES(v1, v2)	YES	YES
lemmatisation	YES (v2)	YES	YES
named entity recognition	NO (in development, v3)	YES	YES
part-of-speech tagging	YES (v2)	YES	YES
entity linking	NO (in development, v3)	YES	YES

3.3 Reference Text for Experiments

We consider, as a sample to compare the tool capabilities and quality of the following text which is taken by a corpus of subjects for exams on Italian as L2. Original text: *Se guardiamo la TV, se leggiamo un giornale o navighiamo in internet, ci vengono continuamente proposte pubblicità di vari prodotti. (sentence 1) La pubblicità è sempre più invadente, ci dà indicazioni su cosa mangiare e bere, su cosa fare quando siamo con gli amici, su quali vacanze preferire, su come vestirci. (sentence 2) Ognuno di noi dovrebbe essere in grado di non farsi condizionare da tutti questi messaggi che cercano di convincerci a comprare e scegliere prodotti commerciali. (sentence 3) Riferisca la Sua opinione a riguardo, spiegando quale influenza ha su di Lei la pubblicità e quale importanza le attribuisce. (sentence 4)*

For the sake of clarity in the figures and the tables, we will focus on sentence 1.

3.4 Dependency Threes and Parsing Information

All the considered tools have rendering functions that visualize the returned syntactic dependency tree associate with sentence processing. Figure 1 shows UDPipe tree while in Fig. 2 SpaCy's and TINT's trees are shown.

UDPipe trees are more user friendly than others because they use a layout which tries giving a visual global picture of the dependency structure. In the

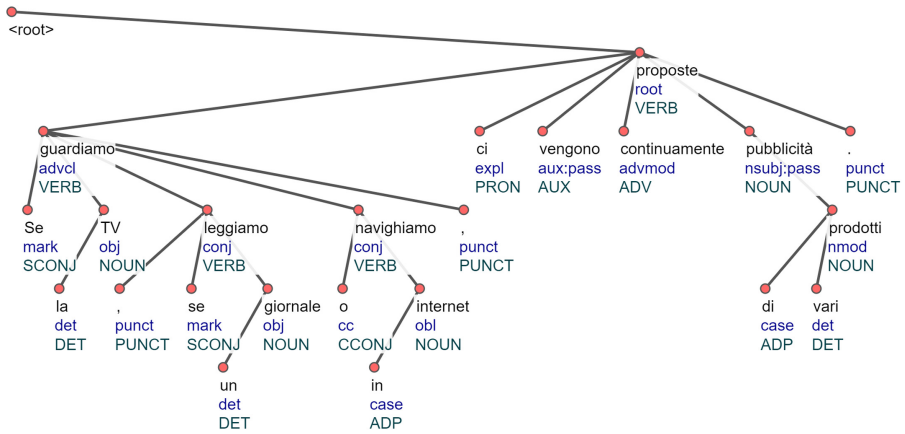

Fig. 1. Dependency tree of sentence 1 by UDPipe2 with *ISDT* treebank

case of TINT and SpaCy, the single line distribution of tokens is more difficult to understand the whole structure. On the other hand, the elements of the other trees are more informative. For instance, UDPipe shows dependency types in the dependant nodes, while both TINT and SpaCy show dependency types as labels overlapping the oriented arcs toward the depending node. TINT also uses colors to provide a glimpse of the POS role of the tokens. Usability of tree visalization is not relevant for automated tasks, where the three is not used by a human but automatically navigated.

The default output of the tools is textual, both for UDPipe and SpaCy, while in the case of TINT the output is provided in JSON format. In the tables in Fig. 3 the different outputs are shown. It is worth noting that UDPipe uses the number of text lines as pointers to define the dependencies, and is rich in information in the nodes (lemma, pos, tags, and elaborate description of mood and verbal form), while SpaCy defaults are very cheap in this regard. The structure of JSON, on the other hand, allows self-describing and easy navigation of the parsing tree, and additional JSON structures for additional information, where the dependency tree description is separated from the node description.

Regarding the quality of parsing, SpaCy fails to correctly analyze Italian sentences in many cases, even in the richer configuration for the Italian language, i.e. using the language package *it_core_news_lg*. For example, the dependency trees and the generated outputs (see Fig. 1, Fig. 2, and Fig. 3) show that even in the case of the sample text, SpaCy is not able to resolve the ambiguity of part of sentence 1. In the expression "vengono continuamente proposte", the word "proposte" is correctly interpreted by UDPipe and TINT as a part of the verbal form where is a verb (VERB V) in past tense form, in English *proposed*, associated with the auxiliary verb "vengono" (AUX V), while SpaCy recognizes "proposte" as nominal subject, in English *proposals*, composed with another noun "pubblicità", i.e. *advertising*. Similar problems of inaccurate parsing of

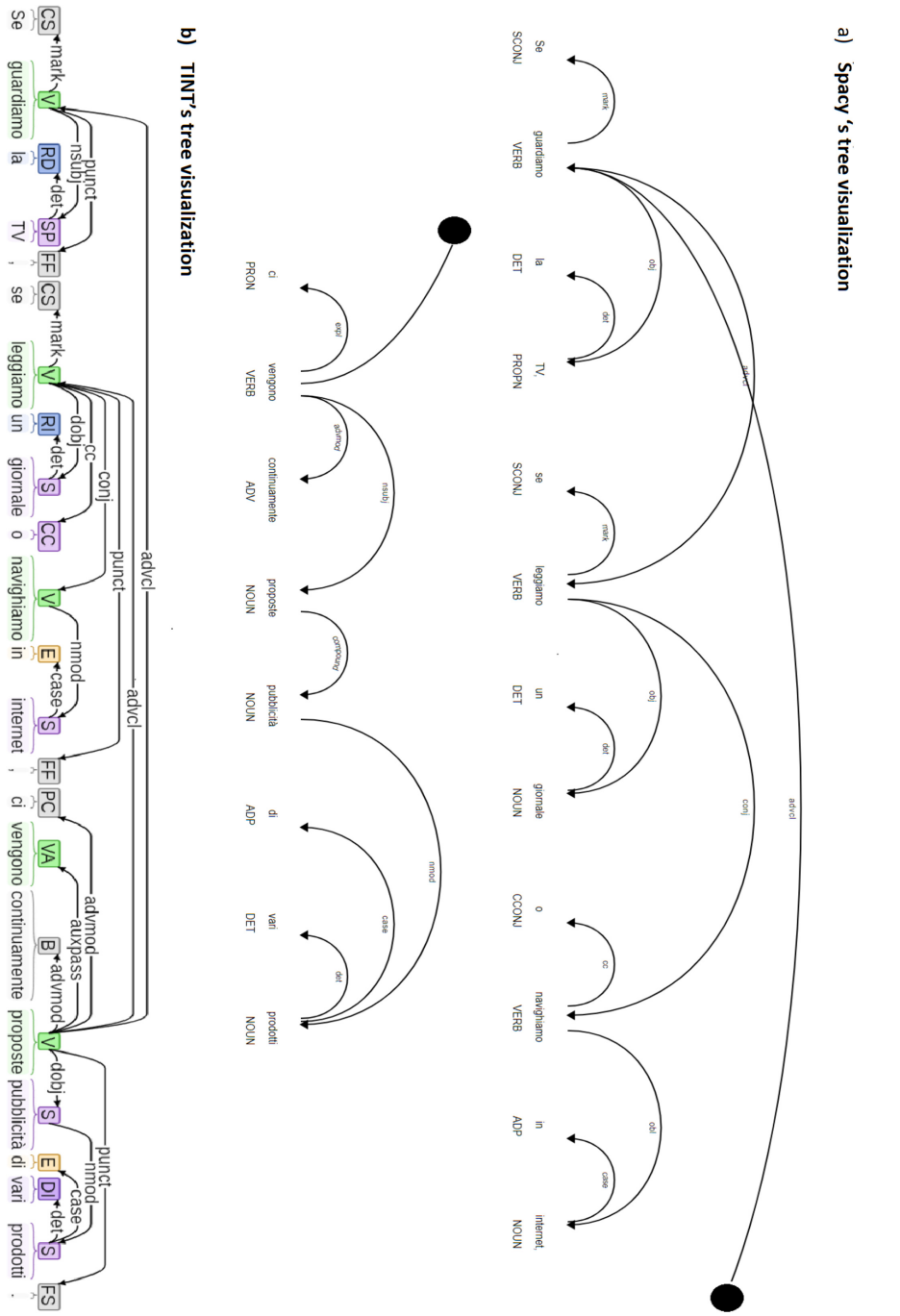

Fig. 2. Dependency tree of sentence 1 by (a) SpaCy *it_core_news_lg* trained pipeline, and (b) TINT

TINT JSON


```
{ "dep": "auxpass",
  "governor": 18,
  "governorGloss": "proposte",
  "dependent": 16,
  "dependentGloss": "vengono"
},
{"index": 16,
  "word": "vengono",
  "originalText": "vengono",
  "lemma": "venire",
  "characterOffsetBegin": 73,
  "characterOffsetEnd": 80,
  "pos": "VA",
"full_morpho": "vengono
venire+v+indic+pres+nil+3+plur"...}
```

```
{ "dep": "advmod",
  "governor": 18,
  "governorGloss": "proposte",
  "dependent": 17,
 "dependentGloss":"continuamente"
},
{"index": 17,
  "word": "continuamente",
"originalText":"continuamente"
  "lemma": "continuamente",
  "characterOffsetBegin": 81,
  "characterOffsetEnd": 94,
  "pos": "B",
"full_morpho": "continuamente
continuamente+adv"...}
```

```
{ "index": 18,
  "word": "proposte",
  "originalText": "proposte",
  "lemma": "proporre",
  "characterOffsetBegin": 95,
  "characterOffsetEnd": 103,
  "pos": "V",
  "full_morpho": "proposte
proporre+v+part+pass+f+nil+p
lur proposta+n+f+plur"



...}
```


UDPipe 2

16	**vengono**	venire	**AUX**	**VA**	18	aux:pass	_
	Mood=Ind|Number=Plur|Person=3|Tense=Pres|VerbForm=Fin						
	TokenRange=73:80						
17	**continuamente**	continuamente	**ADV**	**B** _	18	advmod	_
	TokenRange=81:94						
18	**proposte**	proporre	**VERB**	**V**			
	Gender=Fem|Number=Plur|Tense=Past|VerbForm=Part				0	root	_
	TokenRange=95:103						

Spacy

	text	*lemma*	*pos*	*tag*	*dep*
	vengono	venire	VERB	V	ROOT
	continuamente	continuamente	ADV	B	advmod
	proposte	proposta	NOUN	S	nsubj

Fig. 3. Parsing output of TINT, UDPipe and SpaCy

SpaCy have been observed in several tests with texts from the mentioned corpus, while the accuracy of UDPipe and TINT is confirmed.

4 Conclusions

In this work, we analyzed the main available tools for NLP available for the Italian language, i.e. UDPipe, TINT, and SpaCy. Such tools are considered to be used for automatically performing the pre-processing tasks needed to further analyze texts Italian language, e.g. to assess the phraseological complexity of L2 language learner texts through the use of appropriate complexity measures. All three toolkits offer the basic NLP functions, e.g. tokenization, segmentation, morphological analysis, and lemmatization, while advanced features are currently offered by TINT and SpaCy only, and will be integrated into future releases of UDPipe. Concerning expressivity and usability, in automated tasks,

UDPipe and TINT default outputs are both usable. While the JSON format of TINT seems more portable and usable with other external software tools, UDPipe e SpaCy needs the explicit implementation of a portable format. Regarding quality and reliability, empirical tests done with a corpus of texts for assessing Italian L2 proficiency shows that TINT and UDPipe offer a better level for correctly accomplishing the NLP tasks.

Acknowledgements. This work is partially supported by the Italian Ministry of Research under PRIN Project "PHRAME" Grant n.20178XXKFY.

References

1. Brown, R.: A First Language: The Early Stages. Harvard U. Press, Oxford, England (1973)
2. Bulté, B., Housen, A.: Dimensions of L2 performance and proficiency: complexity, accuracy and fluency in SLA, chapter: defining and operationalising L2 complexity, pp. 21–46. JB (2012)
3. Church, K.W., Hanks, P.: Word association noms, Mutual information, and lexicography. In: Proceedings of the 27th Annual Conference of the Association for Computational Linguistics, vol. 16 no. 1, pp. 22–29 (1989). https://doi.org/10.3115/981623.981633
4. Corazza, M., Menini, S., Cabrio, E., Tonelli, S., Villata, S.: A multilingual evaluation for online hate speech detection. ACM Trans. Internet Technol. **20**(2) (2020). https://doi.org/10.1145/3377323
5. Daller, H., Van Hout, R., Treffers-D Aller, J.: Lexical richness in the spontaneous speech of bilinguals. Appl. Linguist. **24**, 197–222+267 (2003). https://doi.org/10.1093/applin/24.2.197
6. Franzoni, V., Li, Y., Milani, A.: Set semantic similarity for image prosthetic knowledge exchange. In: Misra, S., et al. (eds.) ICCSA 2019. LNCS, vol. 11624, pp. 513–525. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24311-1_37
7. Franzoni, V., Milani, A.: Semantic context extraction from collaborative networks. In: 19th IEEE International Conference on Computer Supported Cooperative Work in Design, CSCWD 2015, Calabria, Italy, May 6–8, 2015, pp. 131–136. IEEE (2015). https://doi.org/10.1109/CSCWD.2015.7230946
8. Franzoni, V., Milani, A.: Structural and semantic proximity in information networks. In: Gervasi, O., et al. (eds.) ICCSA 2017. LNCS, vol. 10404, pp. 651–666. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-62392-4_47
9. https://universaldependencies.org/ : Universal dependencies: comparative statistics of italian treebanks (2021). https://universaldependencies.org/treebanks/it-comparison.html
10. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D.: The Stanford CoreNLP natural language processing toolkit. In: Association for Computational Linguistics (ACL) System Demonstrations, pp. 55–60 (2014)
11. McCarthy, P.M., Jarvis, S.: MTLD, vocd-D, and HD-D: a validation study of sophisticated approaches to lexical diversity assessment. Behav. Res. Methods **42**(2), 381–392 (2010). https://doi.org/10.3758/BRM.42.2.381
12. Menini, S., Tonelli, S., Gasperis, G.D., Vittorini, P.: Automated short answer grading: a simple solution for a difficult task. In: CLiC-it (2019)

13. Milani, A., Spina, S., Santucci, V., Piersanti, L., Simonetti, M., Biondi, G.: Text classification for Italian proficiency evaluation. In: Misra, S., et al. (eds.) ICCSA 2019. LNCS, vol. 11619, pp. 830–841. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24289-3_61
14. Nivre, J., et al.: Universal Dependencies v1: a multilingual treebank collection. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), pp. 1659–1666. European Language Resources Association (ELRA), Portorož, Slovenia (May 2016)
15. Pallotti, G.: A simple view of linguistic complexity. Second Lang. Res. **31**(1), 117–134 (2015). https://doi.org/10.1177/0267658314536435
16. Palmero Aprosio, A., Moretti, G.: Italy goes to Stanford: a collection of CoreNLP modules for Italian. ArXiv e-prints (September 2016)
17. Paquot, M.: Phraseological competence: a missing component in university entrance language tests? insights from a study of efl learners' use of statistical collocations. Lang. Assess. Q. **15**(1), 29–43 (2018). https://doi.org/10.1080/15434303.2017.1405421
18. Paquot, M.: The phraseological dimension in interlanguage complexity research. Second Lang. Res. **35**(1), 121–145 (2019). https://doi.org/10.1177/0267658317694221
19. Polio, C.G.: Second language development in writing: measures of fluency, accuracy, and complexity. kate wolfe-quintero, shunji inagaki, and hae-young kim. honolulu: University of hawai'i press, 1998. pp. viii + 187. Stud. Second Lang. Acquisition **23**(3), 423–425 (2001). https://doi.org/10.1017/S0272263101263050
20. Rescher, N.: Complexity : a philosophical overview / Nicholas Rescher. Transaction Publishers New Brunswick, NJ (1998)
21. Sprugnoli, R., Passarotti, M., Cecchini, F.M., Pellegrini, M.: Overview of the EvaLatin 2020 evaluation campaign. In: Proceedings of LT4HALA 2020–1st Workshop on Language Technologies for Historical and Ancient Languages, pp. 105–110. European Language Resources Association (ELRA), Marseille, France (May 2020)
22. Straka, M.: UDPipe 2.0 prototype at CoNLL 2018 UD shared task. In: Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, pp. 197–207. Association for Computational Linguistics, Brussels, Belgium (October 2018). https://doi.org/10.18653/v1/K18-2020
23. Straka, M., Straková, J.: Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In: Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, pp. 88–99. Association for Computational Linguistics, Vancouver, Canada (August 2017)
24. Zeman, D., et al.: 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. Association for Computational Linguistics, Brussels, Belgium (October 2018). https://doi.org/10.18653/v1/K18-2001
25. About UDPipe. https://ufal.mff.cuni.cz/udpipe (July 2021)