Comparing Link Grammars and Dependency Grammars for parsing German histological reports

Julian Dörenberg[1]

Abstract: The availability of structured data is becoming an increasingly critical factor in medical research. Still, pathologists in Germany document their findings in running text instead of in a structured form. In order to obtain structured data from these report texts, they have to be converted to a more useful form. Link Grammars (LGs) and Dependency Grammars (DGs) both can be used to parse the texts. Hence, LGs and DGs can be used for information extraction on histological reports. This paper aims to compare LGs and DGs, to show why DGs are superior and to evaluate the performance of a DG parser on a corpus of 200 histological reports randomly selected from breast biopsy reports. The DG parser achieved an Unlabelled Attachment Score of 96, a Labelled Accuracy of 95 and a Labelled Attachment Score of 93. Further evaluation shows that the occurrence of medical words which have not been part of the training data does not affect the parsers performance.

Keywords: Natural Language Processing, Text-Mining, AI, Dependency Grammars, Link Grammars, Medical Informatics

1 Introduction

In modern medical research, the availability of structured data is becoming increasingly critical. Data from clinical practice generated from histological reports are of particular interest for cancer research. Although synoptic reporting is on the rise in European medical research [Ka07], pathologists in Germany document most of their findings in running text rather than in a structured form. In order to make the information within these reports available for research, it is critical to convert them into a structured form. One way to perform this information extraction is to use a grammar for a natural language – German here –, to parse the sentences in the reports and to use a downstream application to extract the required information. This paper deals with the comparison of two such grammars, both of which can be obtained by making use of machine learning techniques. After training them, grammars for natural languages typically are used to parse a sentence into a graph. This graph consists of the words within the sentence – its nodes – and grammatical relations between two words in the sentence – its edges. Due to this second aspect, these graphs are also called *(grammatical) relation graphs*.

[1] RWTH Aachen University and University Hospital Aachen,
julian.doerenberg@rwth-aachen.de or jdoerenberg@ukaachen.de

The first grammar type to be discussed here are Link Grammars (LGs). In an LG there is a dictionary that maps each word that is featured in the grammar to a list of possible grammatical contexts that the word can appear in. As these grammars can be denoted in propositional logic in disjunctive normal form, such a grammatical context is called a *disjunct*. Each disjunct contains so-called *connectors*, which denote the option to establish a grammatical relation to a different word within the same sentence.

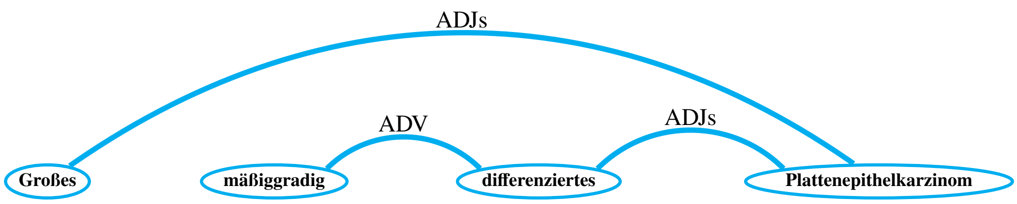
The second grammar type to be discussed here are Dependency Grammars (DGs). DGs also follow the idea that there are grammatical relations between the words, but they differ from LGs in two properties. Firstly, LGs consider the word class – for instance noun or verb –, whereas DGs concentrate on the role a word plays within a sentence – for instance if a noun is the sentences subject or an accusative object. Secondly, the relation graphs generated by a DG are always trees. For LGs, this condition generally does not hold. They allow cycles and - in general - undirected edges.

An example sentence annotated according to both grammar types is given in Figure 1. In the example, the – partial – sentence *Großes mäßiggradig differenziertes Plattenepithelkarzinom* is annotated by using both grammar formalisms. The most important difference is that the relation graphs of DGs are trees whereas there is no hierarchy in LG relation graphs. This results in the property that LGs allow cycles in their relation graphs while DGs do not. The example illustrates two further differences. Firstly, LGs also establish relations to the final dot of a sentence whereas DGs use a root node to model the end of a sentence. Secondly, the tags of the relations denote different linguistic aspects. Discussing these is out of scope here. Intuitively, the tags in DGs denote the grammatical role the word in the child node has in a sentence – and hence takes into account the whole sentence to determine the tag –, while the tag in the LG focuses on the kind of the relationship between the two words which are related to each other.

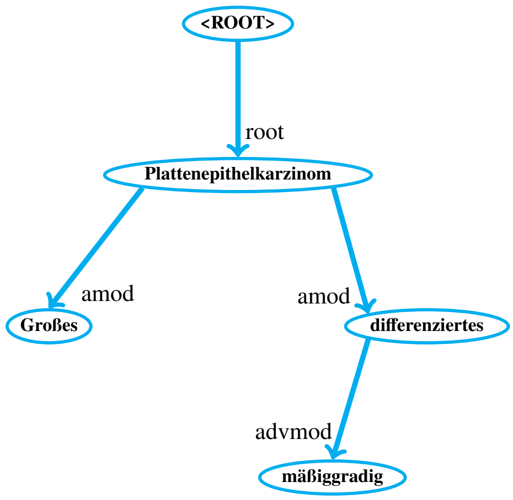
2 Related work

LG were first described in 1996 by Daniel Sleator and Davy Temperley in the context of the English language [ST95]. For linguistic reasons that are outside of the framework of this paper, adaptions to the formalism had to be made in order to make them useable for the German language. This was done by Sandra Kübler in 1998 [Kü02]. For instance, she introduced changes to the connectors, which resulted in the grammatical relation graph being directed. Since then, LGs have been used in multiple information extraction projects on English medical reports. Most of them have since remained in an experimental state and have never been used in real world applications [Zh06]. For the context of the German language, there is no further usage of LGs other than the work of Kübler to the best of my knowledge.

Link Grammar:



Dependency Grammar:



Tab. 1: The Figure shows how the sentence *Großes mäßiggradig differenziertes Plattenepithelkarzinom* is annotated in the Link Grammar formalism – the upper part – and in the Dependency Grammar formalism – the lower part. Each word of the sentence is represented as one node of a relation graph. The edges of the graphs denote the occurrence of a grammatical relations between the words. DGs form trees whereas LGs form non-hierarchical relation graphs.

DGs have their foundation in 1959 when Lucien Tesnieres' work was published posthumously [Te76]. In order to facilitate a modern framework for working with DGs, Christopher Manning, Mihai Surdeanu et al. implemented the Stanford Parser as part of the Stanford CoreNLP pipeline in 2014 [CM14; Ma14]. In 2017, Christopher Manning published a paper together with Timothy Dozat where they trained a recurrent neural network (RNN) to parse sentences by using the DG formalism [DM17; Pa19]. They chose their training data from the Universal Dependencies project [Ma21; Mc13]. Afterwards, they evaluated the model on data from the Universal Dependencies project in different languages. The performance for German was satisfying for use in real world application [DM17]. Dozat and Mannings' neural parser is available for usage with the framework supar, which was developed by Yu Zhang and published in 2018 [Zh18].

Since the first description of DGs, experiments in parsing German medical reports have been done. For instance, Elif Kara, Tatjaa Zeen et al. trained a DG parser on nephrological reports and evaluated its performance by using the Stanford parser [Ka18]. Unfortunately, the parsers performance was too poor to use it in a real world application. Currently, there is a tool in development which utilizes a DG parser based on Dozat and Mannings' work to extract information from histological reports [Dö22]. Figure 1 shows the pipeline executed by this tool.

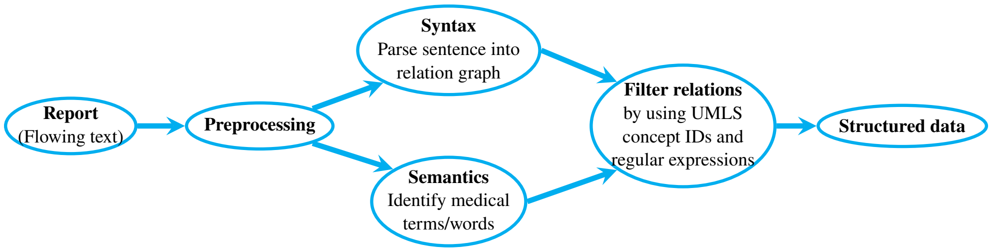


Abb. 1: The pipeline that extracts relations from German histological reports.

After parsing each sentence of the histological report into relation graphs, the tool generates relations of higher arity by attaching relations to each other. For instance, the two relations (*6.5cm*, *groß*) and (*groß*, *Karzinom*) are combined to (*6.5cm*, *groß*, *Karzinom*). Finally, these relations are filtered by using regular expressions and the ontology database UMLS [Bo04] and exported to a csv table. Alternatively to UMLS other ontology databases such as SNOMED CT [St01] can be used. The tool has been evaluated on a data set featuring hepatocellular carcinoma [Dö22]. On this small evaluation it exhibits good performance and is able to extract 98% of the requested information correctly [Dö22]. Table 2 shows their evaluation data.

As given in the table, there is a single error caused by the Dependency Grammar parser. The information *inflammation degree* is not found. This is causes by the – partial – sentence *mit milder entzündlicher Aktivität und portaler sowie septenbildender Fibrose mit Architekturstörung (Grad 2, Stadium 3 nach Desmet)*. As it is syntactically ambiguous whether *Grad 2* refers to *entzündlicher Aktivität*, to *Fibrose* or to *Architekturstörung*, the parser cannot be expected to parse this construct correctly.

# HCCs	**Fibrosis**	**Vascular invasion**	**Tumor diameter**	**Infammation**	**Inflammation degree**	**Distance to reSection area**	**Desmet stage**	**Steatosis**	**Cirrhosis**
1			1,4cm			1mm			TRUE
1		TRUE	5,5cm			0,3cm			FALSE
1	FALSE		4,2cm	FALSE		0,3cm			FALSE
1	TRUE	TRUE	8,5cm	TRUE	—		3		
1			16cm			0,1cm			
1	TRUE	TRUE	4,2cm	TRUE		1,5mm		TRUE	FALSE
1									
1		FALSE	9,5cm			1cm			
1		FALSE	8,5cm	TRUE				TRUE	
1	TRUE		3,6cm			0,2cm	1-2		

Tab. 2: From [Dö22]. The Table shows a dataset extracted by the tool developed in this thesis. Each row represents one report. The column names are information defined to be relevant for researching HCC by the clinic of surgery within the University Hospital Aachen. Information printed in black were extracted correctly. The data can have the two data types boolean, integer or measurement value with respective unit. The Information in the column *Inflammation degree* was extracted wrong, which is denoted by the red dash.

3 Methods

The comparison between LGs and DGs will be purely argumentative. However, the performance of Dozats and Mannings DG parser will be evaluated by following an idea of Carlos Gomez-Rodriguez, Iago Alonso-Alonso and David Vilares [GAV19]. In particular, they recommend to evaluate the performance of a DG parser in two parts.

Firstly, three standard metrics shall be used. For a test corpus, let n_{words} be the number of words in the corpus, n_{rels} be the number of words that correctly have been attached to their parent in the relation graph tree correctly and n_{tags} be the number of words that have been tagged correctly by the parser. Finally, let $n_{n_{rels\_tags}}$ be the number of words that have been attached to their parent in the relation graph tree and have been tagged correctly. Then the three standard metrics are computed as follows: Firstly, The Unlabelled Attachment Score (UAS) is defined as $\frac{n_{rels}}{n_{words}}$. It denotes the proportion of correctly attached grammatical relations to the words. Secondly, the Labelled Accuracy (LA) is defined as $\frac{n_{tags}}{n_{words}}$. It denotes the proportion of correctly tagged words. Thirdly, the Labelled Attachment Score (LAS) is defined as $\frac{n_{rels\_tags}}{n_{words}}$. It denotes the proportion of words being tagged and attached to their parent in the relation graph tree correctly. By definition, LAS combines UAS and LA and it holds $uas \geq las$ and $la \geq las$.

Secondly, the parsing performance shall be evaluated based on the particular use case of the DG parser. In case of the tool presented in Section 2, this is the task of information extraction performed via the pipeline given in Figure 1 where relations of higher arity are generated by attaching relations to each other [Dö22]. Accordingly, this part of the evaluation will be performed by generating the 2-, 3-, and 4-ary relations by using the downstream application. In order to investigate the effect of medical words that were not included in the training data the resulting set of relations will then be filtered for those relations, that contain at least one medical word. The proportion of correctly extracted relations then is used as the evaluation metric.

4 Results

In order to compare LGs and DGs, it is critical to value the linguistic motivation behind both grammar types and investigate properties which firstly affect their parsing performance and secondly their usability.

Both types of grammar model grammatical relations between words. Whereas relation graphs in LGs are non-hierarchical, DGs force their relation graphs to being trees. The problem about non-hierarchical relation graphs is that they allow for cycles, although this is not supported by linguistics. Accordingly, the motivation behind LGs is worse than the one behind DGs, because LGs allow relation graphs which are not supported by linguistics, whereas in DGs this is impossible.

After discussing the linguistic motivation of both grammar types, properties which firstly affect their parsing performance and secondly their usability are compared betwen LGs and DGs. Table 3 shows and compares the properties of LGs and DGs which affect their performance and usability.

Property	LGs	DGs
Lexicalized	Yes	**No**
Adaptations for German required	Yes	**No**
Public Parser for German available	No	**Yes**
Public training data available	No	**Yes**
Neural parser available	No	**Yes**

Tab. 3: Comparison of properties of LGs and DGs. The entry whether one of the grammar types has a property is printed boldly if this grants an advantage over the other grammar type. Kübler implemented an LG parser for German, but due to DG parsers being superior in his evaluation, it has never been published.

The first aspect to be discussed here is lexicalisation. LGs are an example of lexicalized grammars that means that the grammar consists of a dictionary which contains each word known to the grammar. A sentence containing a word that is not part of the grammar cannot be parsed. In opposition to this, DGs are not lexicalised, because they can make use of word embeddings which support unknown words [Bo17]. This makes them superior to LGs, because they are able to handle unknown words. Additionally, typos cannot be recognized by LGs, because the misspelled version of the word is not part of the grammar's dictionary. DGs can bypass this issue by handling the misspelled word as an unknown word.

The second aspect that affects the grammar's parsing performance is the need for specific adaptations of the formalism in order to support the German language. For LGs, Kübler argued that German has a freer word order within the sentences than English. According to its motivation to support the English language, the original formalism strongly preferred creating grammatical relations between words with minimal distance. Hence, Kübler described the changes to the formalism required for the German language [Kü02]. DGs already have been designed based on the idea to parse multilingual texts. Hence, no adaptions are required for their usage in German. This has the advantage that there is no need for an adapted implementation of the parser. As described in Section 2, there are already numerous DG parsers available. For LGs however, this does not hold. Kübler implemented an adapted parsers, but due to the lexicalisation of LGs and her further research on DGs, it has not been published.

Another property affecting the usability of the grammars is the availability of public training data. The Universal Dependencies project has been founded with the aim to annotate corpora with the DG formalism and publish them. Accordingly, numerous corpora have been annotated and published in a number of languages. For instance, there are German corpora based on Twitter data, Google reviews and a number of newspaper articles [In; Mc13]. The data are stored in the CoNLL-U format which simplifies combining multiple corpora [St]. In opposition to this, no training data for German LGs are available to the public to the best of my knowledge.

Data in the CoNLL-U format can simply be used as input data for the training of neural nets. As described in Section 2, modern DG parser implementations are based on neural networks. When training them, it is possible to rely on the advancements the training of neural nets and the availability of the required hardware have made in the past decades. Hence, training can be executed in a very efficient way without the need to implement particular training algorithms. In opposition, no neural parser is available for LGs. In her work, Kübler, however, described a training algorithm that she also implemented. But due to her further work with DGs it has not been published.

In conclusion, DGs are superior to LGs. Hence, it cannot be recommended to use LGs in a real world application. DGs offer a wide range of advantages over LGs and should be preferred.

Accordingly, solely DG parsers are evaluated in this paper. This is done based on the RNN implemented and trained by Dozat and Manning [DM17] which is used in its pretrained version provided by supar [Zh18]. Table 4 shows the evaluation results for the metrics described in Section 2.

Metric	Full corpus [%]	Without localisations [%]
UAS	94	96
LA	92	95
LAS	90	93
2-ary relations	95	97
3-ary relations	91	93
4-ary relations	88	89

Tab. 4: From [Dö22]. Evaluation of the DG Parser trained by Dozat and Manning [DM17]. The metrics described in Section 2 have been computed for two corpora. The corpus including the localisation sentences – left column – consists of 200 sentences, whereas the corpus excluding them – right column – contains 165 sentences.

The model was trained on a number of corpora from the Universal Dependencies project. These include Bulgarian, Catalan, Czech, German, English, Spanish, French, Italian, Dutch, Norwegian, Romanian/Moldavian[1] and Russian corpora [DM17; Mc13]. The evaluation data, however, were explicitely annotated to evaluate Dozat and Manning's model. 200 sentences were randomly selected from breast biopsy reports written by two senior pathologists. Eventually, two sub-corpora of these were evaluated. The left column in Table 4 shows the evaluation results for the full corpus. The right column shows the evaluation results for the same corpus, when sentences on tumor localisation within the affected organ had been removed. Usually, histological reports start with such a sentence – for instance with *links oben außen* – but they do not occur in the further report. From a linguistic point of view, these sentences do not have an unambiguous parsing. In order to not affect the parsing performance, they were removed from the corpus. Afterwards, 165 sentences remained. Overall, the parser showed very good performance on the corpus. In particular, for the most important metric – the UAS – it achieved a score of 96. When restricting the UAS to discard each relation that does not contain a medical word, the score for the 2-ary-relations is obtained. This was computed as 97. This demonstrates that the occurrence of medical words does not affect the parsing performance negatively, although these words have neither been included in the training data, nor in the dictionary of the words embeddings.

Medical words sometimes occur as Multi Word Expressions (MWEs), such as *Carcinoma in situ* for instance. Semantics of MWEs are determined from the combination of the words they contain. In the given example *Carcinoma* does not have a semantics on its own in German. It is the Latin word for the German word *Karzinom*. The whole expression *Carcinoma in situ* however has a semantics in German. It is reasonable to assume that medical MWEs reduce the parsing performance, because neither the whole expression nor the separate medical words are featured in the training data. Hence, it is sensible to set up the hypothesis that the DG parser will not parse them correctly. This hypothesis has turned

[1] Using the ISO 639-1 code it remains ambiguous whether the data include sentences in Romanian, Moldavian or both and Dozat and Manning did not elaborate on that in their paper.

out to be true: In the few MWEs that occur in the 200 breast biopsy sentences, neither of the MWEs has been parsed correctly. In each case either the grammatical relation or its type was assigned incorrectly to at least one of the separate words. Unfortunately, the corpus is too small to allow for a proper analysis of the errors and to find a pattern behind the parsing errors.

5 Conclusion

LGs have shown to be inferior to DGs on a conceptual level for two reasons. Firstly, they are lexicalized and hence can neither handle unknown words nor typos. DGs resolve this problem by supporting word embeddings [DM17; Pa19; Zh18]. Complementing this issue, the set of linguistic properties modelled in LGs is incomplete: The underlying formalism allows for cycles in the relation graph, although this is impossible from a linguistic point of view. This applies to both, the German as well as the English language. In conclusion, it is not recommended to use LGs for any kind of application, but it is preferable to use DGs instead.

In order to investigate multiple DG parsers, the framework supar can be used [Zh18]. One of these parsers pretrained by Dozat and Manning [DM17] shows good results in parsing histological reports on the chosen corpus of 200 sentences randomly selected from breast biopsy reports. However, the corpus is quite small, thereby limiting the evaluation of the parsing performance on histological reports. Hence, more data will be annotated and the evaluation of the extension is ongoing. Currently, there is evidence that the DG parser lacks in precision when parsing sentences which contain MWEs. Hence, the training data corpus of Dozat and Manning will be extended by adding sentences that contain medical MWEs. Afterwards, Dozats and Mannings training will be repeated on the enlarged corpus. The goal behind this is to increase the performance of the DG parser when parsing MWEs and identify the reason why they cause the parser to parse them incorrectly.

Nevertheless, it has been shown that Dependency Grammar parsers can already be helpful in their current state of development, even without above modifications [Dö22]. But if the advances described above are successful and the DG parser's performance will be verified on a larger evaluation data corpus, DGs will eventually significantly advance the parsing of medical reports.

Literatur

[Bo04] Bodenreider, O.: The Unified Medical Language System (UMLS): integrating biomedical terminology. en, Nucleic Acids Research 32/90001, S. 267D–270, Jan. 2004, ISSN: 1362-4962, URL: https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkh061, Stand: 09. 11. 2021.

[Bo17] Bojanowski, P.; Grave, E.; Joulin, A.; Mikolov, T.: Enriching Word Vectors with Subword Information. arXiv:1607.04606 [cs]/, arXiv: 1607.04606, Juni 2017, URL: http://arxiv.org/abs/1607.04606, Stand: 15. 11. 2021.

[CM14] Chen, D.; Manning, C. D.: A fast and accurate dependency parser using neural networks. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). S. 740–750, 2014.

[DM17] Dozat, T.; Manning, C. D.: Deep Biaffine Attention for Neural Dependency Parsing. arXiv:1611.01734 [cs]/, März 2017, URL: http://arxiv.org/abs/1611.01734, Stand: 28. 10. 2021.

[Dö22] Dörenberg, J.: Extraction of HCC-related data from histological reports by using a Dependency Grammar, English, Presentation at 105. Jahrestagung der Deutschen Gesellschaft für Pathologie, Münster, Germany, Juni 2022, URL: https://www.cbmb.ukaachen.de/wp-content/uploads/2022/06/DGP2022_Praesentation_Doerenbergetal.pdf.

[GAV19] Gómez-Rodríguez, C.; Alonso-Alonso, I.; Vilares, D.: How important is syntactic parsing accuracy? An empirical evaluation on rule-based sentiment analysis. en, Artificial Intelligence Review 52/3, S. 2081–2097, Okt. 2019, ISSN: 0269-2821, 1573-7462, URL: http://link.springer.com/10.1007/s10462-017-9584-0, Stand: 28. 10. 2021.

[In] Ines Rehbein and Josef Ruppenhofer and Bich-Ngoc Do: tweeDe – A Universal Dependencies treebank for German tweets, URL: https://aclanthology.org/W19-7811.pdf.

[Ka07] Karim, R. Z.; Van Den Berg, K. S.; Colman, M. H.; McCarthy, S. W.; Thompson, J. F.; Scolyer, R. A.: The advantage of using a synoptic pathology report format for cutaneous melanoma: Synoptic pathology reports in melanoma. en, Histopathology 52/2, S. 130–138, Dez. 2007, ISSN: 03090167, URL: http://doi.wiley.com/10.1111/j.1365-2559.2007.02921.x, Stand: 02. 05. 2021.

[Ka18] Kara, E.; Zeen, T.; Gabryszak, A.; Budde, K.; Schmidt, D.; Roller, R.: A Domain-adapted Dependency Parser for German Clinical Text. In: A Domain-adapted Dependency Parser for German Clinical Text. Verlag der Österreichischen Akademie der Wissenschaften, 0xc1aa5576_0x003a12bd, 2018, ISBN: 978-3-7001-8437-9, URL: https://hw.oeaw.ac.at/8437-9, Stand: 24. 01. 2022.

[Kü02] Kübler, S.: Learning a Lexicalized Grammar for German. New Methods in Language Processing and Computational Natural Language Learning/, 2002.

[Ma14] Manning, C. D.; Surdeanu, M.; Bauer, J.; Finkel, J. R.; Bethard, S.; McClosky, D.: The Stanford CoreNLP natural language processing toolkit. In: Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations. S. 55–60, 2014.

[Ma21] de Marneffe, M.-C.; Manning, C. D.; Nivre, J.; Zeman, D.: Universal Dependencies. en, Computational Linguistics/, https://universaldependencies.org/, S. 1–54, Mai 2021, ISSN: 0891-2017, 1530-9312, URL: https://direct.mit.edu/coli/article/doi/10.1162/coli_a_00402/98516/Universal-Dependencies, Stand: 14. 11. 2021.

[Mc13] McDonald, R.; Nivre, J.; Quirmbach-Brundage, Y.; Goldberg, Y.; Das, D.; Ganchev, K.; Hall, K.; Petrov, S.; Zhang, H.; Täckström, O. et al.: Universal dependency annotation for multilingual parsing. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). S. 92–97, 2013.

[Pa19] Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; Chintala, S.: PyTorch: An Imperative Style, High-Performance Deep Learning Library. In (Wallach, H.; Larochelle, H.; Beygelzimer, A.; Alché-Buc, F. d.; Fox, E.; Garnett, R., Hrsg.): Advances in Neural Information Processing Systems. Bd. 32, Curran Associates, Inc., 2019, URL: https://proceedings.neurips.cc/paper/2019/file/bdbca288fee7f92f2bfa9f7012727740-Paper.pdf.

[St] Stenstrom, E.: CoNLL-U, URL: https://github.com/EmilStenstrom/conllu.

[St01] Stearns, M. Q.; Price, C.; Spackman, K. A.; Wang, A. Y.: SNOMED clinical terms: overview of the development process and project status. eng, Proceedings. AMIA Symposium/, S. 662–666, 2001, ISSN: 1531-605X.

[ST95] Sleator, D.; Temperley, D.: Parsing English with a Link Grammar. CoRR abs/cmp-lg/9508004/, 1995.

[Te76] Tesnière, L.: Éléments de syntaxe structurale. Klincksieck, Paris, 1976, ISBN: 978-2-252-01861-3.

[Zh06] Zhou, X.; Han, H.; Chankai, I.; Prestrud, A.; Brooks, A.: Approaches to text mining for clinical medical records. In: Proceedings of the 2006 ACM symposium on Applied computing - SAC '06. ACM Press, Dijon, France, S. 235, 2006, ISBN: 978-1-59593-108-5, URL: http://portal.acm.org/citation.cfm?doid=1141277.1141330, Stand: 01. 05. 2021.

[Zh18] Zhang, Y.: Supar, 2018, URL: https://github.com/yzhangcs/parser.