Web-based Annotation Interface for Derivational Morphology

Lukáš Kyjánek
Charles University, Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics
Prague, Czechia
kyjanek@ufal.mff.cuni.cz

Abstract

The paper presents a visual interface for manual annotation of language resources for derivational morphology. The interface is web-based and created using relatively simple programming techniques, and yet it rapidly facilitates and speeds up the annotation process, especially in languages with rich derivational morphology. As such, it can reduce the cost of the process. After introducing manual annotation tasks in derivational morphology, the paper describes the new visual interface and a case study that compares the current annotation method to the annotation using the interface. In addition, it also demonstrates the opportunity to use the interface for manual annotation of syntactic trees. The source codes are freely available under the MIT License on GitHub.

1 Introduction

Making manual annotations is a common task when a high-quality language resource is created. The more complex the annotation task, the more time consuming it is for an annotator, an expert in a linguistic field captured by the resource. Consequently, the cost of creating such resource can be high. The simplest approach is to simplify the task; however, it is not possible in many cases.

This paper presents such task and how to approach it in the field of annotating language resources of derivational morphology. When annotating derivational data, annotators must make many decisions at once. This complexity leads not only to the prolongation of the annotation process but also to mistakes that must be additionally re-annotated. As the simplification of these decisions is out of the question, a freely available web-based visual interface has been created to make the annotation process easier for annotators. The fact that annotating derivational morphology using the interface is faster than the currently used annotation method is also validated by real annotators.

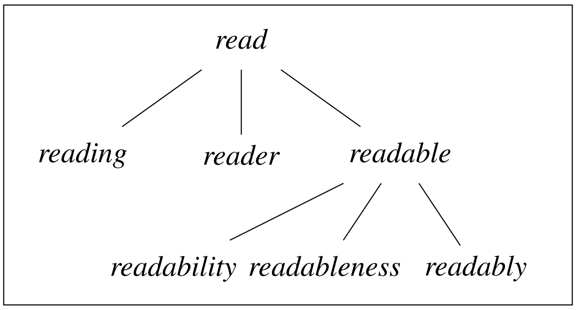

Figure 1: A derivational family of the verb *read*.

The paper is structured as follows. Section 2 describes the current state of annotating derivational morphology. Section 3 focuses on the new interface as well as its applicability to other annotation tasks. Section 4 provide a comparison of the current annotation method *versus* the utilisation of the interface. Section 5 concludes the paper.

2 Related Work

2.1 Linguistic Background

Morphological derivation is a process of forming new lexemes by modifying the already existing ones. For instance, the noun *reader* is derived by attaching the lexical affix *-er* to the morphological base of the verb *read*. Štekauer et al. (2012) document this process across many world languages.

One of the widely known approaches to derivational morphology (cf. Dokulil 1962; Buzássyová 1974; Horecký et al. 1989; Furdík 2004; Štekauer 2005) models all derivationally related lexemes (DERIVATIONAL FAMILY) on the basis of:

(i) a system of directly derivationally related lexemes grouped around a single base lexeme, e.g., *read* > *read-ing*, *read-er*, and *read-able*;

(ii) a sequence of consecutive derivatives, e.g., *read* > *read-able* > *read-abil-ity*.

If these parts are applied recursively to a single underived lexeme, it results in derivational families structured in rooted trees, see Figure 1.

Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: System Demonstrations, pages 10 - 16
July 10-15, 2022 ©2022 Association for Computational Linguistics

2.2 Data Resources

There is a lot of lexical resources of derivational morphology (cf. Kyjánek 2018),[1] many of which model derivational families in rooted trees, concurring with the above-mentioned theory.[2] Most of the other existing resources that capture derivational families in non-tree-shaped data structures have been harmonised into the rooted trees and are available in the Universal Derivations collection (Kyjánek et al., 2020; Kyjánek et al., 2021).[3]

The reasons why some resources do not model derivational families in rooted trees cover a whole range from technical to theoretical reasons.[4] For example, DErivBase for German (Zeller et al., 2013) has been created by exploiting so-called *derivational rules* extracted from grammars in a form of sophisticated regular expressions which the authors have utilised to search derivationally related lexemes in a given lexeme set. Consequently, the resulting resource violates the main constraint of the rooted tree structure that each lexeme can have at most one base lexeme, e.g., the adjective *glatt* (*smooth*) and the verb *glätten* (*to smooth*) are captured as bases for the noun *Glätte* (*smoothness*). The manual annotation is thus necessary not only before the creation of a high-quality resource but also after that for its harmonisation, for example.

2.3 Manual Annotation Process

Annotators in the field of derivational morphology have to make many small decisions at once, even if they are only supposed to annotate Boolean decision like whether a given derivational relation is acceptable, e.g., *glatt* (*smooth*) > *Glätte* (*smoothness*) vs. *glätten* (*to smooth*) > *Glätte* (*smoothness*). To fulfil the conditions of the linguistic approach described in Section 2.1, they must decide (i) whether a given derived lexeme is really a derivative; if yes, then (ii) from which base lexeme it is derived; and (iii) whether the final decision does not violate constraints of the rooted tree structure with other derivationally related lexemes; and (iv) whether they decide consistently across derivational families. As these questions are interrelated, their simplification seems to be out of the question.

```
+    glatt_A       Glätte_Nf
+    glatt_A       glätten_V
+    glätten_V     glättend_A
-    glätten_V     Glätte_Nf
```

Figure 2: Example of a common .tsv file format for manual annotation. The data is stored in columns (annotator's mark, base lexeme, and derivative; lexemes are equipped with part-of-speech tags: `A` for adjectives, `V` for verbs, `Nf` for feminine nouns).

One of the common ways of making manual annotation of derivational relations is to list them in a file and assign each of them with a mark representing the presence/absence of the relation in the resulting rooted tree, see Figure 2. The annotation task seems easy if the data for annotation is small; however, the data is relatively large in practice. For instance, you can see manual annotations of one thousand relations from Wiktionary annotated before their addition into DeriNet for Czech.[5] Moreover, individual derivational families can be relatively large, especially in languages with rich derivational morphology, which even complicates the annotation process.

2.4 Tools for Linguistic Annotation

To the best of our knowledge, there is no available tool for making manual annotation of derivational morphology. The previous cases have relied on either non-public software developed solely for the annotation project or on simple text-based methods. There are few visualisation tools that at least display the data, e.g., WFL explorer[6] (Passarotti and Mambrini, 2012), DeriNet viewer[7] (Žabokrtský et al., 2016), DeriSearch v1[8] and v2[9] (Vidra and Žabokrtský, 2017, 2020), and Canoonet.[10] However, none of them allows editing the data.

There are also no case studies for annotation of derivational morphology neither in the ACL Anthology nor in the recent Handbook of Linguistic Annotation (Ide and Pustejovsky, 2017). However, the handbook and other similar cases from

[1] Perhaps the earliest modern case of a large-scale resource is CELEX2 (Baayen et al., 1995) with its annotations of derivational morphology of Dutch, English, and German.

[2] For example: DeriNet for Czech (Vidra et al., 2021), Spanish (Faryad, 2021), Persian (Haghdoost et al., 2019), and Russian (Kyjánek et al., 2021), Polish and Spanish Word-Formation Networks (Mateusz et al., 2018a,b), and Word Formation Latin (Litta et al., 2016).

[3] https://ufal.mff.cuni.cz/universal-derivations

[4] They are described in the text on the harmonisation.

[5] https://github.com/vidraj/derinet/blob/master/data/annotations/cs/2018_04_wiktionary/hand-annotated/der0001-1000.tsv

[6] http://wfl.marginalia.it/

[7] https://ufal.mff.cuni.cz/derinet/derinet-viewer

[8] https://ufal.mff.cuni.cz/derinet/derinet-search

[9] https://quest.ms.mff.cuni.cz/derisearch2/v2/databases/

[10] https://www.lehrerfreund.de/schule/1s/online-grammatik-canoo/2319

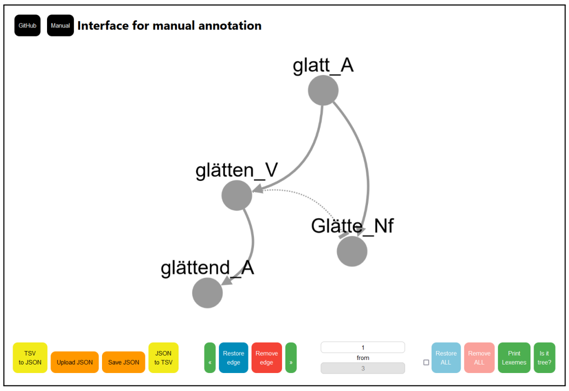

Figure 3: Screenshot of the freely available online Interface for manual annotation of derivational morphology. It captures the same derivational family as Figure 2.

the fields of annotating data resources of syntax (cf. Tyers et al. (2017)) and inflectional morphology (cf. Obeid et al. (2018); Alosaimy and Atwell (2018)) suggest that visualisation of the derivational data and a possibility to edit it in the visualisation interface is one of the best options to facilitate the annotation process.

3 Interface for Manual Annotations

The new interface should visualise the annotated data (cf. Figure 2) and allow annotators to edit it, leading to the facilitation of the annotation process. In this case, the process should lead to the data of derivational morphology annotated into the tree-shaped structure presented in Section 2.1. The functionality and design of the interface have already been tested by real annotators. They provided feedback for re-designing the interface to the current state, see Figure 3. Validation of the interface is presented in Section 4.

The development of such tools however never ends, as many things can be still improved. Therefore, the interface is distributed as open-source (on GitHub)[11] with a detailed manual that includes also example data. It is possible to run the interface both locally and online.[12]

3.1 Technical Properties

The interface is made as a web application thanks to which annotators can run it in any web browser, e.g., Microsoft Edge, Google Chrome, Mozilla Firefox, and Safari. It prevents problems with users' installation of any programming environments. The interface is responsive by default (without implementing the interface separately for individual types of devices), so the annotators can annotate not only on their computers but also on touchscreens of their mobile devices. The interface is programmed using HTML5, CCS3, and JavaScript with the libraries jQuery, CytoScape.js, and Notify.js. These technologies runs the interface on the client's side/user's device that brings the mention benefit, but, on the other hand, the interface suffers from the limits of web browsers.[13]

```
[
 {
  "nodes":
   [
    {"data":
      {"name":"glättend_A","id":"glättend_A"}
    },
    {"data":
      {"name":"Glätte_Nf","id":"Glätte_Nf"}
    },
    {"data":
      {"name":"glatt_A","id":"glatt_A"}
    },
    {"data":
      {"name":"glätten_V","id":"glätten_V"}
    }
   ],
  "edges":
   [
    {"data":
      {"target":"glatt_A","source":"Glätte_Nf",
       "intoTree":"solid"}
    },
    {"data":
      {"target":"glatt_A","source":"glätten_V",
       "intoTree":"solid"}
    },
    {"data":
      {"target":"glätten_V","source":"Glätte_Nf",
       "intoTree":"dotted"}
    },
    {"data":
      {"target":"glätten_V","source":"glättend_A",
       "intoTree":"solid"}
    }
   ]
  }
]
```

Figure 4: Input and output .json format of the annotation data for the interface from Figure 3.

As for the input/output file formats, the interface is ready to process derivational families stored in the .json format; see Figure 4. There is also a function implemented to convert the .tsv file into the .json file and *vice versa* (the `TSV_to_JSON` and `JSON_to_TSV` buttons).

3.2 Design and Functionality

The interface screen consists of top and bottom button bars and a central canvas for data visualisation

[11] https://github.com/lukyjanek/uder-annotation-interface

[12] https://lukyjanek.github.io/subpages/uder-annotation-interface/UDerAnnotation.html

[13] For instance, Google Chrome has a memory limit of 512MB for 32-bit and 1.4GB for 64-bit systems. However, it is still enough memory for common manual annotation.

and annotation. While the top bar includes support buttons, such as a link to the source codes stored on GitHub and manual, the bottom bar contains buttons for manual annotation.

The data is loaded using the `Upload_JSON` button. The annotator can zoom in/out the screen and move the displayed nodes and their relations. Positions of nodes on canvas are stored in the .json file. Annotators can thus return quickly to already annotated derivational families. During the annotation process, they select relations to be annotated and change their state by one of the following buttons: `Restore_edge` (draws the relation by a solid line representing that it should be present in the resulting family) or `Remove_edge` (dotted line, should be absent). The annotators can switch between families using the green arrow buttons or the text box. At the end of the annotation, the work is saved with the `Save_JSON` button.

Several functionalities have been added based on the annotators' feedback. If the annotator writes a word or its substring to the text box, the interface searches for the family containing the word/substring and visualises it. To facilitate the annotation of families with many relations, there are two buttons that remove and restore all derivational relations in the displayed derivational family. For annotators, it is sometimes easier to remove all edges and build such a large tree from scratch by restoring individual edges. Some buttons list all lexemes from the visualised family and check whether the solid lines in the annotated family are organised in a rooted tree. In addition, keyboard shortcuts have been introduced for all the functions. After all these changes, the annotators confirmed that the interface makes their work easier and faster.

3.3 Applicability to Different Tasks

To show the robustness of the new interface for annotation of data in tree-shaped structures, a brief experiment with annotating syntactic data has been performed. The harmonised syntactic data from the Universal Dependencies collection (Zeman et al., 2021) was selected for this experiment. The only thing in need was to create a script that would convert the input .conllu format into the .json format required by the annotation interface. Figure 5 displays the German sentence '*Absolut empfehlenswert ist auch der Service.*' (*The service is also highly recommended.*) from the corpus GSD (McDonald et al., 2013) in the annotation interface.

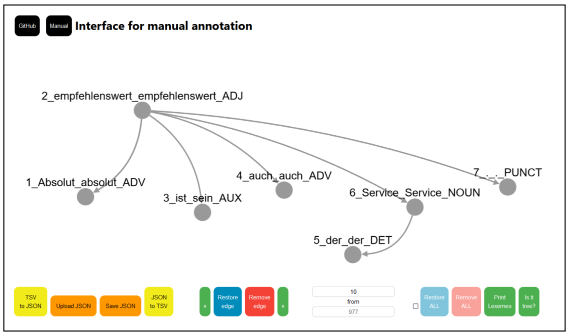

Figure 5: Screenshot of the freely available online Interface for manual annotation of derivational morphology applied to the data from syntactic treebank from Universal Dependencies. The underscores separate id, token, lemma, part-of-speech category.

4 Human Validation

To validate the usefulness of the newly created interface for manual annotation of derivational morphology, as described in the Section 2.3, a simple annotation experiment has been done with human annotators. Two methods of manual annotation are compared: (a) the currently used method when annotators work in the traditional text processor with the .tsv format, and (b) the annotation by using the newly created visual interface with the .json format. The main expectation is that annotating the same data by using the interface should be faster. The individual parts of this experiment, such as the input sample as well as the annotated ones, are stored on the GitHub repository with the source codes.

4.1 Annotation Experiment

The experiment involved 12 human annotators (university students of other than linguistic studies). They all annotated the same sample of derivational families; however, six of the annotators did it in the text processor with the .tsv file format, i.e., the currently used method of annotation such data, while the other six annotators used the newly created visual interface with the .json file format.

The annotators were instructed to annotate the given data in a such way that it concurs with the approach to model derivational families in rooted trees (Section 2.1), i.e., that each lexeme can have at most one base lexeme and that the morphological complexity should grow from the root to the leaves. They also got the instructions related to the individual annotation methods, e.g., all functionalities and buttons of the interface were explained to the annotators who would use the interface.

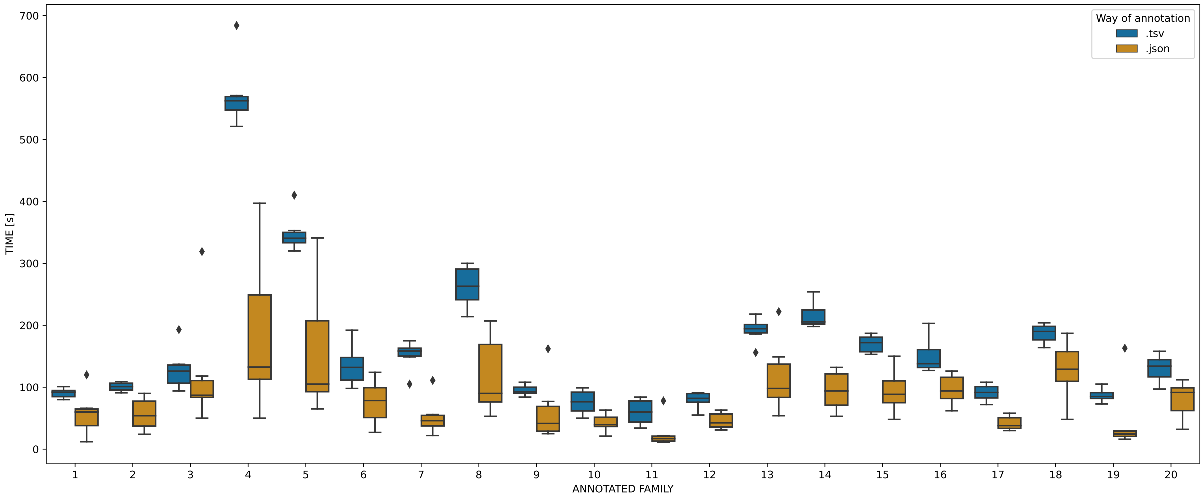

Figure 6: Time spent by annotators on annotating individual 20 derivational families in the .tsv file format and using the new visual interface with the .json file format.

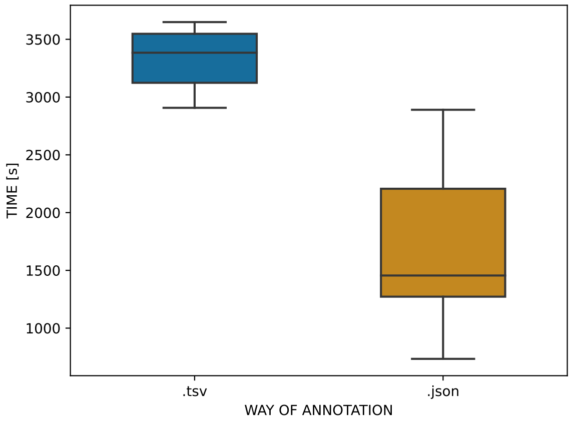

Figure 7: Total time spent by annotators on annotating 20 derivational families in the .tsv file format and using the new visual interface with the .json file format.

4.2 Annotation Sample

The annotation sample consisted of two similar sets of ten different derivational (sub-)families selected from Czech DeriNet with respect to their numbers of lexemes, derivational relations, depths (morphological complexity) of the original trees, and the part-of-speech categories of the tree roots. Each set of the sample thus includes four families with the noun and verbal tree roots and two families with the adjective tree roots; it has the following ranges: for the number of lexemes from 6 to 20, for relations from 6 to 24, for depth from 2 to 5.

A few random incorrect connections (from 2 to 5 relations) were made in all families in the sample. The annotators were supposed to annotate these errors and let the other correct relations.

The division of the sample into two sets of ten families can provide an overview of how the annotators' experience with the annotation using the assigned method influence the time spent on the annotation of individual families. The assumption associated with this is that the time spent over the second set should be less than for the first set because the annotators learn the annotation process with each annotated family. On the other hand, if the visual interface is useful, then the time spent on annotating the .json file format by using the interface should be still lower than on the annotation of the .tsv file format.

As for the specific properties of the individual families in the sample, the smallest families in terms of the number of lexemes are numbered as 1, 2, 9, 10, 11, 12, 19, 20; and the families 9 and 19 includes a complete graph which the annotators have to annotate into the rooted tree structure.

4.3 Results

The main hypothesis that the annotation process is faster if the newly created visual interface is used (with .json format) instead of the current annotation method (with .tsv format) was proved, at least for Czech, a language that has rich derivational morphology. Annotators with the visual interface annotated faster in the case of all annotated derivational families; see Figure 6. However, the difference in time was small for smaller families, which indicates that the current .tsv annotation method is comparably good as the annotation using the interface in the case of derivational families with few tens of relations. If the family is bigger, then the annotators were much faster when using the visual interface. In total, Figure 7 illustrates that the annotation process with visual interface takes noticeably fewer seconds than the currently used annotation method.

The secondary hypothesis that the annotators are faster in the second half of the sample, especially when this half shares the same parameters in terms of numbers of lexemes and relations, was not proved so conclusively as the main hypothesis. There is such trend in the second half of the sample, but the differences are not so radical.

5 Conclusion

When developing high-quality data, especially data that contains more complicated structures, developers often ask for manual annotations. They need the annotations when they create, extend, test, or evaluate the data. Annotation of complex phenomena is time-consuming and increases data production costs. Therefore, it seems worth spending time to simplify the annotation process.

In this paper, a case study about manual annotation of complex phenomena from derivational morphology has been presented. As a way of simplifying the annotation process, a web-based visual annotation interface in which annotators can edit the displayed data has been created. The interface is freely available (cf. Footnote 11). It was created in direct collaboration with several annotators who tested the interface on data of real derivational families and provided useful feedback. The annotators have also rated the annotation process with the created interface as more attractive, easier, and faster, which led to greater savings of time (and potentially money spent on the development of the resulting resource while still achieving high quality). Their feedback has led to the addition of several new functionalities, such as keyboard shortcuts and the button for checking treeness, that significantly speed up the annotation process. In addition, the desired benefits of the interface have been validated by annotators, and the paper describes this validation. It confirms that usage of the new interface rapidly speeds up the annotation process compared to the current method of annotating data for derivational morphology.

In general, this paper underlines that a tool/interface can be created by relatively basic techniques but can still save a lot of annotators' time and effort. One of the crucial points is, however, to be open-minded and to communicate with annotators. Since the annotators know how they must think during the annotating, they can specify their needs and provide informed feedback. There is still (and always will be) a lot of ways in which such interface can be improved or extended; they remain for future work. The important message is that even a simple interface can greatly facilitate the manual annotation process. While a programmer creates such interface in a few hours, annotators can save days of work.

Acknowledgements

This work was supported by the Grant No. GA19-14534S of the Czech Science Foundation, and the Grant No. START/HUM/010 of Grant schemes at Charles University (reg. No. CZ.02.2.69/0.0/0.0/19_073/0016935). It was using language resources developed, stored, and distributed by the LINDAT/CLARIAH-CZ project.

References

Abdulrahman Alosaimy and Eric Atwell. 2018. Web-based Annotation Tool for Inflectional Language Resources. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*, pages 3933–3939.

Harald R. Baayen, Richard Piepenbrock, and Leon Gulikers. 1995. CELEX2. Linguistic Data Consortium, Catalogue No. LDC96L14.

Klára Buzássyová. 1974. *Sémantická struktúra slovenských deverbatív*. Veda, Bratislava.

Miloš Dokulil. 1962. *Tvoření slov v češtině 1: Teorie odvozování slov*. Academia, Prague.

Ján Faryad. 2021. DeriNet.ES 0.6. Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University; included in the UDer collection.

Juraj Furdík. 2004. *Slovenská slovotvorba*. NÁUKA, Prešov.

Hamid Haghdoost, Ebrahim Ansari, Zdeněk Žabokrtský, and Mahshid Nikravesh. 2019. DeriNet.FA 0.5. Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University; included in the UDer collection.

Ján Horecký, Klára Buzássyová, Ján Bosák, et al. 1989. *Dynamika slovnej zásoby súčasnej slovenčiny*. Veda, Bratislava.

Nancy Ide and James Pustejovsky. 2017. *Handbook of Linguistic Annotation*, 1st edition. Springer Publishing Company, Incorporated.

Lukáš Kyjánek, Zdeněk Žabokrtský, Jonáš Vidra, and Magda Ševčíková. 2021. Universal Derivations v1.1. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Lukáš Kyjánek, Zdeněk Žabokrtský, Magda Ševčíková, and Jonáš Vidra. 2020. Universal Derivations 1.0, A Growing Collection of Harmonised Word-Formation Resources. *The Prague Bulletin of Mathematical Linguistics*, 115:5–30.

Lukáš Kyjánek. 2018. Morphological Resources of Derivational Word-Formation Relations. Technical Report TR-2018-61, Faculty of Mathematics and Physics, Charles University.

Lukáš Kyjánek, Olga Lyashevskaya, Anna Nedoluzhko, Daniil Vodolazsky, and Zdeněk Žabokrtský. 2021. DeriNet.RU 0.5. Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University; included in the UDer collection.

Eleonora Litta, Marco Passarotti, and Chris Culy. 2016. Formatio Formosa est. Building a Word Formation Lexicon for Latin. In *Proceedings of the 3rd Italian Conference on Computational Linguistics*, pages 185–189.

Lango Mateusz, Magda Ševčíková, and Zdeněk Žabokrtský. 2018a. Polish Word-Formation Network 0.5. Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University; included in the UDer collection.

Lango Mateusz, Magda Ševčíková, and Zdeněk Žabokrtský. 2018b. Spanish Word-Formation Network 0.5. Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University; included in the UDer collection.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal Dependency Annotation for Multilingual Parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria. Association for Computational Linguistics.

Ossama Obeid, Salam Khalifa, Nizar Habash, Houda Bouamor, Wajdi Zaghouani, and Kemal Oflazer. 2018. MADARi: A web interface for joint Arabic morphological annotation and spelling correction. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*.

Marco Passarotti and Francesco Mambrini. 2012. First Steps towards the Semi-automatic Development of a Wordformation-based Lexicon of Latin. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 852–859.

Pavol Štekauer. 2005. Onomasiological Approach to Word-Formation. In Pavol Štekauer and Rochelle Lieber, editors, *Handbook of Word-Formation*, pages 207–232. Springer, Dordrecht.

Pavol Štekauer, Salvador Valera, and Lívia Körtvélyessy. 2012. *Word-Formation in the World's Languages: A Typological Survey*. Cambridge University Press, New York.

Francis Tyers, Mariya Sheyanova, and Jonathan Washington. 2017. UD Annotatrix: An annotation tool for Universal Dependencies. In *Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories*, pages 10–17.

Jonáš Vidra, Zdeněk Žabokrtský, Lukáš Kyjánek, Magda Ševčíková, Šárka Dohnalová, Emil Svoboda, and Jan Bodnár. 2021. DeriNet 2.1. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University; included in the UDer collection.

Jonáš Vidra and Zdeněk Žabokrtský. 2017. Online Software Components for Accessing Derivational Networks. In *Proceedings of the Workshop on Resources and Tools for Derivational Morphology (DeriMo 2017)*, pages 129–139. EDUCatt.

Jonáš Vidra and Zdeněk Žabokrtský. 2020. Next Step in Online Querying and Visualization of Word-Formation Networks. In *Proceedings of the 23rd International Conference on Text, Speech and Dialogue (TSD 2020)*, pages 144–152. Springer.

Britta Zeller, Jan Šnajder, and Sebastian Padó. 2013. DErivBase: Inducing and Evaluating a Derivational Morphology Resource for German. In *ACL*, volume 1, pages 1201–1211. Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).

Daniel Zeman, Joakim Nivre, Mitchell Abrams, et al. 2021. Universal Dependencies 2.9. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Zdeněk Žabokrtský, Magda Ševčíková, Milan Straka, Jonáš Vidra, and Adéla Limburská. 2016. Merging Data Resources for Inflectional and Derivational Morphology in Czech. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1307–1314.