LEA – Linguistic Exercises with Annotation tools

Fabian Barteld
Institute of German Studies
Universität Hamburg
fabian.barteld@uni-hamburg.de

Johanna Flick
Institute of German Studies
Heinrich Heine University Düsseldorf
flick@uni-duesseldorf.de

Abstract

In this paper we present LEA (Linguistic Exercises with Annotation tools). LEA is a new didactic concept helping students to become familiar with corpus linguistic methods and annotation tools. The main idea behind LEA is that classical linguistic exercises are being solved with annotation tools. We will present the advantages of this method (e.g. didactic benefits, automatic correction) and describe two already existing LEA e-learning packages: part-of-speech annotation using tab-separated-value files with spreadsheet software and syntactic analysis with Synpathy.[1]

1 Introduction

Corpus linguistics as a method has become more and more important for linguistic research in the last decades (Gries, 2009; Bender and Good, 2010). However, creating or working with digitally annotated data can be a difficult task for beginners. This is especially true within the philologies where most students lack the computational expertise needed for corpus linguistics (Bubenhofer, 2011). In addition, there is often not much time to teach the related methods within these disciplines. This leads to the situation that students have to deal with a lot of practical issues when doing their first own empirical research: "Working empirically is a complex task and acquiring the necessary technical expertise in order to use the tools can lead to frustration" (Bubenhofer, 2011, p. 148)[2].

With LEA (Linguistic Exercises with Annotation tools) we introduce an e-learning approach that helps solving this problem. On the one hand, LEA is a collection of ready-to-use exercises that can be incorporated into classes when linguistic categories like part of speech (PoS) are being introduced. On the other hand, LEA is more than just an e-learning package. It is a didactic methodology which helps students to become familiar with corpus linguistic methods and tools, since the exercises come in the shape of annotation tasks: Students solve traditional linguistic exercises like PoS categorization or syntactic analysis and at the same time learn how to use annotation tools.

In the first part of the paper, we explain the concept behind LEA. Then, this concept is further illustrated with the description of two already existing LEA packages (PoS classification and syntactic analysis) which have been designed for first semester introductory courses to German linguistics as part of German philology. In the last part, we present the experiences we made from applying LEA in teaching.

LEA is available at https://korpuslab.github.io/lea.

2 LEA – The concept

LEA combines traditional linguistic exercises with annotation tools, simplified annotation guidelines, and tools for correction and evaluation. The concept can be best explained in terms of the "Trojan Horse metaphor" except that there is something good coming out from the horse's inside, i.e. the computational know how: While doing exercises students learn about corpus linguistic methods and ways of creating sustainable annotated data, since they are basically solving an annotation task. The data that is annotated consists of invented sentences as they are typically used in class for illustrating and practicing specific linguistic concepts. The students simply use annotation tools to "write down" their solution for the exercise. Therefore, LEA differs from other didactic approaches where either existing corpus resources are used (Beißwenger

[1] Both authors contributed equally to the paper.

[2] Own translation of: "Empirisches Arbeiten ist aufwändig und wenn gleichzeitig noch die technischen Fertigkeiten erlangt werden müssen, um die Werkzeuge anwenden zu können, kann dies zu Frustgefühlen führen."

and Storrer, 2011) or a corpus resource is created by students (Zeldes, 2017).

Using annotation tools for exercises has several advantages over the classical way of using pen and paper. First of all, it exposes students to software and concepts from corpus linguistics before they actually try to employ corpus linguistic methods. The students can benefit from this new knowledge in their later research and do not have to struggle with the annotation tools and file formats when acquiring statistical methods and other tools that are necessary for corpus linguistic research. Secondly, as the students' answers are created digitally in a certain format, it is easy for the lecturer to evaluate the answers and prepare the discussion in class. Thirdly, LEA supports media change: Instead of solving exercises with pen and paper the students use special computer programs. The new ways of visualization can support a better understanding of linguistic concepts. In addition, the exercises come with annotation guidelines so that the concepts that the students have learned and which they are practicing are being presented from a new perspective. This also supports the understanding process.

Each LEA package has the same structure. It consists of the following parts: (i) the exercise, (ii) a manual describing how to work with the exercise plus examples and annotation guidelines, (iii) a manual for the annotation tool including information on how to obtain it, (iv) the sample solution, and (v) a tool for automatic correction and evaluation of the answers. The whole package comes as hypertext allowing to navigate the manuals and get the files for the exercises.

With the help of the manuals students learn to install and use the software needed for the exercises. Note that this step does not necessarily need to be supervised in class since the manuals are built as self-learning resources. Especially courses with a full schedule and lack of time for technical issues benefit from this approach. As mentioned above, LEA aims at linguistics as part of philologies where students as well as instructors often lack computational expertise. Therefore, the manuals lead users with screenshots through the steps that are necessary to solve the exercise. They also contain additional background information which gives students the opportunity to learn more about the tools and the data format. More experienced learners can skip what they already know.

LEA packages can be used straight away as they come but they can also be adapted to the needs of the course. How to create new exercises is explained in the manuals. Due to its modular structure, that is to say modules describing an annotation tool are separated from modules containing the exercise, it is easy to add a new exercise: The exercise and its description can simply be combined with already existing technical parts (mainly the description of the annotation tool).

Up to now, there are two LEA packages, one for part of speech and one for phrase types and syntactic functions.

3 The part-of-speech exercise

Every student of linguistics is being confronted with PoS exercises at some point of her academical studies, usually in the first year. A classical task would be assigning categories like noun, verb or preposition to specific word forms, e.g. the tokens in the sentence *The cat sits under the table*. Since not every classification is as easy as this example a lot of practice is necessary.

The LEA PoS package provides a new form for this traditional task: Instead of annotating the words with pen and paper the exercise comes in a tab-separated-value (tsv) file that can be edited with common spreadsheet applications like Libre Office Calc[3]. We chose this as annotation tool for the exercise because we assume that students are usually already familiar with spreadsheet software which makes it easier for them to fulfill the annotation task. Furthermore, the students learn how to work with a typical file format for tabular data, a format that is often used for data in corpus linguistics[4]. When solving the exercise they learn about various aspects connected to this file format, e.g. handling different kinds of text delimiters like tab or comma, and choosing the right encoding when opening text files which contain non-ASCII characters like German umlauts.

When classifying the examples, the students are asked to use a tagset. For the created exercise, we use a simplified version of the Stuttgart-Tübingen Tagset (STTS) (Schiller et al., 1999), which is a defacto standard for PoS tagging for German (Zinsmeister et al., 2014). It is for example used in one of the annotation layers in the German Reference

[3] https://www.libreoffice.org/

[4] Compare the various variants of the CoNLL formats for example the CoNLL-U format used by the universal dependencies project (http://universaldependencies.org/format.html).

Corpus (DeReKO) (Belica et al., 2009), which is an important resource for the study of contemporary German.

On a more conceptual level, the method introduces the concept of tagsets and annotation guidelines to the students. While the introduction of PoS in linguistics often emphasizes the theoretical aspects of the categories (e.g. morphological vs. syntactic criteria), annotation guidelines focus on distinguishing the categories in real texts. These different approaches complement each other and help the students to understand the different parts of speech.

The students use a slightly modified version of the original STTS as a tagset: Some categories, e.g. different kinds of particles, were converted into broader classes, since they are not being distinguished in most of the introduction literature to German linguistics (Linke et al., 2004; Meibauer et al., 2007; Busch and Stenschke, 2014) or syntax (Dürscheid, 2012; Pittner and Berman, 2015). Some tags like for instance "FM" for foreign-language material were omitted entirely. On the other hand, we extended STTS with categories that are usually distinguished in courses for German linguistics; we split for example the category "article" ("ART") into "definite article" ("ARTDEF") and "indefinite article" ("ARTINDEF").

The exercise sentences and the categorization of the tokens itself have been created on the basis of the above mentioned introductions to German syntax. As such, the exercise should be usable in a wide range of courses for German linguistics. However, changing the exercise to the needs of a specific course is as easy as creating a new tsv-file and – if necessary – adapting the description of the tagset which is presented as a html-file created from Markdown, a simple markup language[5].

Solving the exercises digitally by creating tsv-files allows to facilitate the correction and evaluation of the students' solutions by automatically comparing them to the sample solution. In order to make this possible for lecturers who themselves do not have a background in corpus linguistics, we created a simple tool. It is written in Java and should therefore run on all common platforms. The tool reads in a sample solution and a folder containing the students' solutions. It then compares each solution to the sample solution and creates a version where deviations from the sample solution are marked and the correct answer is provided for each of the students' solution. To help the lecturer getting an insight on frequent mistakes, i.e. problematic categories, the tool also outputs helpful statistics (e.g. mean number of errors, most frequent mistakes and a confusion matrix) as is illustrated in Fig. 1.

4 The syntax exercise

The syntax exercise focuses on phrase types (e.g. nominal phrase or prepositional phrase) and syntactic functions (subject, object etc.) which represent the basic inventory of a wide rage of linguistic theories. As annotation tool for the exercise we chose *Synpathy*[6], a tool for manual syntactic annotation. Even though Synpathy is rather old and not under active development anymore, there are three reasons why we preferred it to other programs:

Firstly, current tools that allow syntactic annotation, e.g. Arborator (Gerdes, 2013) and WebAnno (Yimam et al., 2016), use dependencies, a theoretical framework which only plays a secondary role in German linguistics. Since Synpathy allows to annotate constituents with labeled and crossing edges it is suitable for the kind of syntactic analysis that we find in most of the established introduction literature (Dürscheid, 2012; Pittner and Berman, 2015, among others). The presented syntax exercise does not aim at presenting analyses based on a specific theoretical framework like dependency or phrase-structure but to introduce general categories like phrase types and syntactic functions. It is also possible to include other categories like topological fields using Synpathy, cf. the annotation guidelines for Tüba-D/Z (Telljohann et al., 2015) where topological fields are annotated.

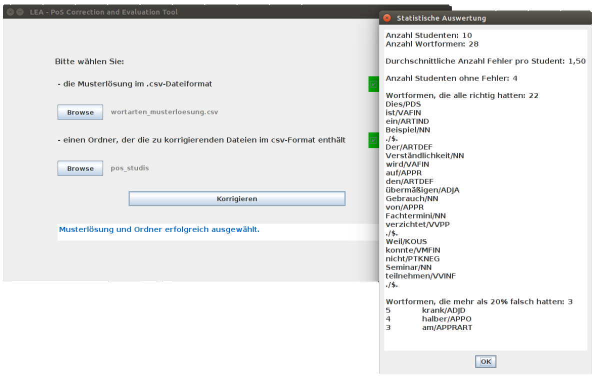
Secondly, Synpathy is relatively easy to use especially when compared to general purpose annotation tools like GATE (Cunningham et al., 2011) and Atomic (Druskat et al., 2014) which could also be used to annotate constituency trees but are harder to learn due to their general nature and therefore are not suitable for the purpose of LEA.

Thirdly, Synpathy saves the annotation in the Tiger-XML-format[7], which is well supported by other tools. There are for example importers for

[5] https://daringfireball.net/projects/markdown/syntax

[6] https://tla.mpi.nl/tools/tla-tools/older-tools/synpathy/

[7] http://www.ims.uni-stuttgart.de/forschung/ressourcen/werkzeuge/TIGERSearch/doc/html/TigerXML.html



LEA - PoS Correction and Evaluation Tool

Bitte wählen Sie:

- die Musterlösung im .csv-Dateiformat

Browse wortarten_musterloesung.csv

- einen Ordner, der die zu korrigierenden Dateien im csv-Format enthält

Browse pos_studis

Korrigieren

Musterlösung und Ordner erfolgreich ausgewählt.

Statistische Auswertung

```
Anzahl Studenten: 10
Anzahl Wortformen: 28

Durchschnittliche Anzahl Fehler pro Student: 1,50

Anzahl Studenten ohne Fehler: 4

Wortformen, die alle richtig hatten: 22
Dies/PDS
ist/VAFIN
ein/ARTIND
Beispiel/NN
./$.
Der/ARTDEF
Verständlichkeit/NN
wird/VAFIN
auf/APPR
den/ARTDEF
übermäßigen/ADJA
Gebrauch/NN
von/APPR
Fachtermini/NN
verzichtet/VVPP
./$.
Weil/KOUS
konnte/VMFIN
nicht/PTKNEG
Seminar/NN
teilnehmen/VVINF
./$.

Wortformen, die mehr als 20% falsch hatten: 3
5          krank/ADJD
4          halber/APPO
3          am/APPRART
```

Figure 1: Correction tool

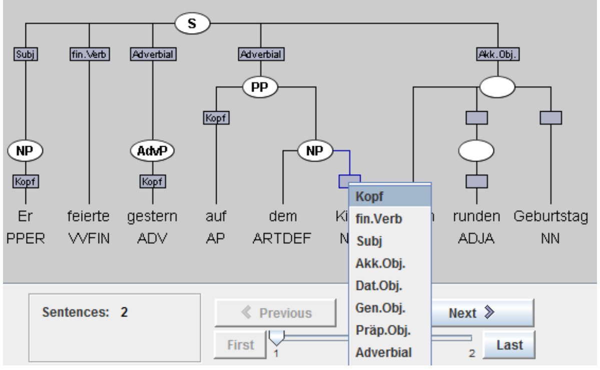


Figure 2: Syntax annotation

the corpus search tool ANNIS (Krause and Zeldes, 2016). Thus, students can use Synpathy to create syntactically annotated data for their own studies that can be searched and visualized with ANNIS.[8]

We want to highlight a didactic aspect of using Synpathy to solve the syntax exercises in particular: Students often have problems distinguishing between parts of speech, syntactic functions and phrase types. By using Synpathy, they are confronted with a visualization of an analysis that clearly distinguishes the three types of categories, cf. Fig. 2 which illustrates how the sentence *Er feierte gestern auf dem Kiez seinen runden Geburtstag.* (Engl.: 'He celebrated his big birthday on the Kiez yesterday') would be annotated. While the phrase types are represented as labels of oval nodes, the syntactic functions are edge labels that are visualized in rectangles. The PoS tags are shown below the tokens. This helps students to differentiate between those category types. Note that the analyis in Fig. 2 is based on the simplified version of STTS created for the exercise. AP stands for Adposition. This tag corresponds to the more specific tags APPR for Preposition and APPO for Postposition in STTS. The two other tags from STTS that are subordinated under AP – APPRART for preposition-article contractions and APZR for the right part of circumpositions – do not have a counterpart in our simplified tagset as these special cases do not appear in the exercise.

For the automatic correction, we extended Synpathy with a function to compare a set of Tiger-XML files (the students' solutions) with the currently opened file (the sample solution).[9] The comparison of the trees is done using the algorithm given in Brants and Skut (1998). The results are presented in tabular form, giving an overview over wrongly analyzed sentences.

5 Experiences with LEA

In order to assess the didactic value and the usability of the two LEA packages we conducted surveys in three introductory courses for German linguistics during summer term 2016 and again in three courses during winter term 2016/2017 using LimeSurvey[10]. The feedback was used for improvements of the packages.

The overall reaction to LEA was positive and the packages are still in use in the current summer term. The survey indicates that working with the annotation guidelines contributed to a better understanding of the linguistic concepts as described in the previous sections: when asked whether STTS was helpful for understanding parts of speech, all of the students agreed.

Furthermore, the testing phase of LEA revealed some problems regarding the usage of the annotation tools: with the first version many students were not able to open the exercise and solve it properly. For example, the Tiger-XML files used by Synpathy were opened in the browser when the students followed the link to the file in the manual. Many students did not know how to save the file to disk and open it with Synpathy. To circumvent this problem, we packed the exercise file into a zip-archive which does not open in the browser. A different type of problem appeared when the tab-separated-value files were opened with Excel directly and not via the option to import the data "from text" since this leads Excel to expect ";" as separator resulting

[8] However, getting the data into ANNIS is not trivial and would have to be taught in a later course.

[9] This extended version of Synpathy can be found at https://github.com/fab-bar/Synpathy.

[10] https://www.limesurvey.org/de/

in a corrupted file (the first column contains the tab used as separator). Even though the students were able to solve the exercise in this case, the automatic evaluation did not work.

These observations confirm the lack of technical know how as described by Bubenhofer (2011) and at the same time stress the need for didactic concepts like LEA.

6 Conclusion and further work

With LEA we present both a collection of e-learning packages and an underlying didactic concept which helps students to become familiar with annotation tools and methods from corpus linguistics. In this paper, we described two exercises for syntax related categories that introduce tsv-files for parts of speech and Synpathy as an annotation tool for phrase types and syntactic functions. They are currently used in introductory courses for German linguistics at Universität Hamburg and Heinrich Heine University Düsseldorf and under active development in the sense that we integrate solutions to problems that are encountered.

In the future, we are planning to extend LEA with new exercises. While the current packages focus on syntax, exercises for a wider range of topics are possible, e.g. annotation of semantic roles with SALTO (Burchardt et al., 2006), spoken language with EXMARaLDA (Schmidt and Wörner, 2009) or phonetic analysis with Praat (Boersma and Weenink, 2013).

While these programs – similar to spreadsheet applications and Synpathy used for the presented exercises – are all specialized standalone tools, many currently developed tools are web-based, e.g. WebAnno. Web-based tools have the advantage that annotators do not need to install the software and annotation tasks may be prepared for them, e.g. in form of a project with the annotation categories needed as well as the data which is supposed to be annotated. In this case, the students only need to learn how to annotate with the tool. It makes the task easier, because they do not have to work with tsv- or xml-files. However, this way the students will not learn how to deal with such file formats – one of the learning targets of the presented LEA exercises. Without this knowledge students will always depend on the availability of the data and tools in a web-based application. Yet, at some point they will need to import or export data, e.g. for further analysis of the annotated data using spreadsheet software. Therefore, we also plan to create exercises where the students will have to prepare and import data into a web-based tool like WebAnno themselves.

Acknowledgments

The creation of the presented LEA packages were funded by the "Innovationsfonds für Studium und Lehre" (Universität Hamburg, Fakultät für Geisteswissenschaften) and assisted by Christiane Höltmann and Alexandra Lindt.

We would like to thank Melanie Andresen, Yvonne Hettler, Eleonore Schmitt, Renata Szczepaniak, Annika Vieregge and Heike Zinsmeister for testing LEA in their courses and all the students who supplied feedback. We also thank the anonymous reviewers for their helpful remarks.

References

Michael Beißwenger and Angelika Storrer. 2011. Digitale Sprachressourcen in Lehramtsstudiengängen: Kompetenzen – Erfahrungen – Desiderate. *Journal for Language Technology and Computational Linguistics (JLCL)*, 26(1):119–139.

Cyril Belica, Marc Kupietz, Andreas Witt, and Harald Lüngen. 2009. The morphosyntactic annotation of DeReKo: Interpretation, opportunities, and pitfalls. In Marek Konopka, Jacqueline Kubczak, Christian Mair, František Šticha, and Ulrich H. Waßner, editors, *Grammatik und Korpora 2009. Dritte Internationale Konferenz. Mannheim, 22.-24.9.2009*, number 1 in Korpuslinguistik und interdisziplinäre Perspektiven auf Sprache (CLIP). Narr, Tübingen.

Emily M. Bender and Jeff Good. 2010. A grand challenge for linguistics: Scaling up and integrating models. *White paper contributed to NSF's SBE 2020 initiative*.

Paul Boersma and David Weenink. 2013. Praat: doing phonetics by computer. `http://www.praat.org/`.

Thorsten Brants and Wojciech Skut. 1998. Automation of treebank annotation. In *Proceedings of the Joint Conferences on New Methods in Language Processing and Computational Natural Language Learning*, pages 49–57.

Noah Bubenhofer. 2011. Korpuslinguistik in der linguistischen Lehre: Erfolge und Misserfolge. *Journal for Language Technology and Computational Linguistics (JLCL)*, 26(1):141–156.

Aljoscha Burchardt, Katrin Erk, Anette Frank, Andrea Kowalski, Sebastian Padó, and Manfred Pinkal. 2006. SALTO – a versatile multi-level annotation tool. In *Proceedings of the Fifth International*

Conference on Language Resources and Evaluation (LREC2006).

Albert Busch and Oliver Stenschke. 2014. *Germanistische Linguistik: Eine Einführung*. Narr, Tübingen, 3rd edition.

Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damljanovic, Thomas Heitz, Mark A. Greenwood, Horacio Saggion, Johann Petrak, Yaoyong Li, and Wim Peters. 2011. *Text Processing with GATE (Version 6)*.

Stephan Druskat, Lennart Bierkandt, Volker Gast, Christoph Rzymski, and Florian Zipser. 2014. Atomic: An open-source software platform for multi-level corpus annotation. In *Proceedings of the 12th Konferenz zur Verarbeitung natürlicher Sprache (KONVENS 2014)*, pages 228–234.

Christa Dürscheid. 2012. *Syntax: Grundlagen und Theorien*. Vandenhoeck & Ruprecht, Göttingen et al., 6th edition.

Kim Gerdes. 2013. Collaborative Dependency Annotation. In *Proceedings of the Second International Conference on Dependency Linguistics (DepLing 2013)*, pages 88–97.

Stefan Th. Gries. 2009. What is corpus linguistics? *Language and Linguistics Compass*, 3(5):1–17.

Thomas Krause and Amir Zeldes. 2016. ANNIS3: A new architecture for generic corpus query and visualization. *Literary and Linguistic Computing*, 31(1):118–139.

Angelika Linke, Markus Nussbaumer, and Paul R. Portmann. 2004. *Studienbuch Linguistik*. Number 121 in Reihe Germanistische Linguistik. Niemeyer, Tübingen, 5th edition.

Jörg Meibauer, Ulrike Demske, Jochen Geilfuß-Wolfgang, Jürgen Pafel, Karl Heinz Ramers, Monika Rothweiler, and Markus Steinbach. 2007. *Einführung in die germanistische Linguistik*. Metzler, Stuttgart, 2nd edition.

Karin Pittner and Judith Berman. 2015. *Deutsche Syntax: Ein Arbeitsbuch*. Narr, Tübingen, 6th edition.

A. Schiller, S. Teufel, C. Stöckert, and C. Thielen. 1999. Guidelines für das Tagging deutscher Textcorpora mit STTS. Technical report, Universities of Stuttgart und Tübingen.

Thomas Schmidt and Kai Wörner. 2009. EXMARaLDA – creating, analysing and sharing spoken language corpora for pragmatic research. *Pragmatics*, 19(4):565–582.

Heike Telljohann, Erhard W. Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. 2015. Stylebook for the Tübingen treebank of written German (TüBa-D/Z). Technical report, Seminar für Sprachwissenschaft, University of Tübingen.

Seid Muhie Yimam, Silvana Hartmann, Iryna Gurevych, Anette Frank, and Chris Biemann. 2016. A web-based tool for the integrated annotation of semantic and syntactic structures. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH)*, pages 76–84.

Amir Zeldes. 2017. The GUM corpus: creating multilayer resources in the classroom. *Language Resources and Evaluation*, 51(3):581–612.

Heike Zinsmeister, Ulrich Heid, and Kathrin Beck. 2014. Adapting a part-of-speech tagset to non-standard text: The case of STTS. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC14)*, pages 4097–4104.