Linguistic Corpus Annotation - Project Summary

Matteo Brivio, Pia Schwarz

February 26, 2021

1 Overview

The **Karlsruhe Children's Text** corpus (KCT) and the **H2, E2, ERK1 Children's Writing** corpus (H2) are among the largest collections of German children's writing.
At the moment, both corpora are not in a computationally-readable format and store transcriptions of the collected data, word- and sentence-level annotations as well as meta-data, on the same layer.
The KCT corpus contains 1,701 texts elicited from students from grade 1 through 8, while the H2 corpus counts a total of 2,117 texts. For both corpora the obtained texts have been digitised in two forms: the original text, referred to as *achieved*, and the intended text, where all spelling errors have been removed, referred to as *target*.
Both KCT and H2 appear to be using the same annotation conventions to mark mistakes at both word and sentence level. On the other hand, there seems to be a slight mismatch with respect to the meta-data included in both corpora. Nonetheless, relevant information such as date of collection, age, gender, grade and language spoken at home are common to both corpora.

2 Goal

The goal is to bring the data into a more universal format which is suitable for a linguistic corpus: PAULA XML. All available meta-data as well as the existing transcriptions will be encoded in such a way that the original corpus is recoverable. Two base layers will be created, a normalized layer which is created from the target text. This layer represents the children's writings without annotations and corrected for spelling mistakes. The second layer is the achieved text with the resolved annotation symbols which ensures that the original writing of the child is conserved.
Additionally, two more kinds of linguistic annotation will be added: one annotation contains all error annotation symbols and is attached to the achieved layer, and a part of speech annotation attached to the normalized layer. Both layers and both annotations will be generated automatically.
The final corpus encoded with PAULA XML should be convertible with the

Pepper tool, so it can be imported and queried in ANNIS or further annotated in tools like WebAnno.

3 Project Steps

From our present point of view the following steps are necessary to realize the project (work split is indicated in brackets):

- Agree on a corpus model and research possible encoding formats (e.g. TIGER xml and PAULA xml) **(Matteo and Pia)**
- Parse and collect data (achieved, target, metadata) from the source text files of the corpus **(Matteo)**
- Create function to tokenize the parsed data (achieved, target) **(Pia)**
- Create a function to resolve annotation symbols (e.g. und{2} → und und), returning the units to build the achieved layer **(Pia)**
- Create a function to resolve annotation symbols (e.g. und{2} → und) returning the units to build the normalized layer **(Pia)**
- Create a function to spellcheck tokens of the target layer and catch potential typos, thus improving the result of further automatic annotation steps e.g. POS tagging **(Matteo)**
- Create a function to automatically POS tag the normalized layer **(Matteo)**
- Create functions to generate all of the xml files required by the PAULA format **(Matteo)**
- Create a function generating corpus-level metadata **(Pia)**
- Create a function converting original .jpg scanned documents into .pdf and include them into the corpus **(Matteo)**
- Create a function collecting and integrating all of the above to encode the corpus **(Matteo, Pia)**
- Convert the the corpus with Pepper and inspect it with ANNIS **(Matteo, Pia)**
- Debug scripts and mock corpora examples and final encoded corpus **(Matteo, Pia)**

4 Tools

Given the current goal and steps we plan on using the following set of tools:

- Python
- lxml to generate all the XML documents
- Pepper and ANNIS
- SpaCy (`https://spacy.io/`)
- Spellchecker Hunspell (`http://hunspell.github.io/`)
- img2pdf 0.4.0