CorA: A web-based annotation tool for historical and other non-standard language data

Marcel Bollmann, Florian Petran, Stefanie Dipper, Julia Krasselt
Department of Linguistics
Ruhr-University Bochum, 44780 Bochum, Germany
{bollmann|petran|dipper|krasselt}@linguistics.rub.de

Abstract

We present CorA, a web-based annotation tool for manual annotation of historical and other non-standard language data. It allows for editing the primary data and modifying token boundaries during the annotation process. Further, it supports immediate retraining of taggers on newly annotated data.

1 Introduction[1]

In recent years, the focus of research in natural language processing has shifted from highly standardized text types, such as newspaper texts, to text types that often infringe orthographic, grammatical and stylistic norms normally associated with written language. Prime examples are language data produced in the context of *computer-mediated communication* (CMC), such as Twitter or SMS data, or contributions in chat rooms. Further examples are data produced by learners or historical texts.

Tools trained on standardized data perform considerably worse on "non-standard varieties" such as internet data (cf. Giesbrecht and Evert (2009)'s work on tagging the web or Foster et al. (2011)'s results for parsing Twitter data) or historical language data (Rayson et al., 2007; Scheible et al., 2011). This can mainly be attributed to the facts that tools are applied out of domain, or only small amounts of manually-annotated training data are available.

A more fundamental problem is that common and established methods and categories for language analysis often do not fit the phenomena occurring in non-standard data. For instance, grammaticalization is a process of language evolution where new parts of speech are created or words switch from one class to another. It is difficult to draw strict categorial boundaries between words that take part in a continuous smooth transition of categories. Factors like these can also affect the way the data should be tokenized, along with other problems such as the lack of a fixed orthography.

In the light of the above, we developed a web-based tool for manual annotation of non-standard data. It allows for editing the primary data, e.g. for correcting OCR errors of historical texts, or for modifying token boundaries during the annotation process. Furthermore, it supports immediate retraining of taggers on newly annotated data, to attenuate the problem of sparse training data.

CorA is currently used in several projects that annotate historical data, and one project that analyzes chat data. So far, about 200,000 tokens in 84 texts have been annotated in CorA. Once the annotation process is completed, the transcriptions and their annotations are imported into the ANNIS corpus tool (Zeldes et al., 2009) where they can be searched and visualized.

The paper focuses on the annotation of historical data. Sec. 2 presents the tool, and Sec. 3 describes the data model. Sec. 4 concludes.

2 Tool Description

CorA uses a web-based architecture:[2] All data is stored on a server, while users can access and edit annotations from anywhere using their web browser. This approach greatly simplifies collaborative work within a project, as it ensures that all users are working on the same version of the data at all times, and requires no software installation on the user's side. Users can be assigned to individual project groups and are only able to access documents within their group(s).

2.1 The annotation editor

All annotation in CorA is done on a token level; the currently supported annotation types are part-

[1]The research reported here was financed by Deutsche Forschungsgemeinschaft (DFG), Grant DI 1558/5-1.

[2]It implements a standard AJAX architecture using PHP 5, MySQL, and JavaScript.

Proceedings of the 8th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH) @ EACL 2014, pages 86–90,
Gothenburg, Sweden, April 26 2014.

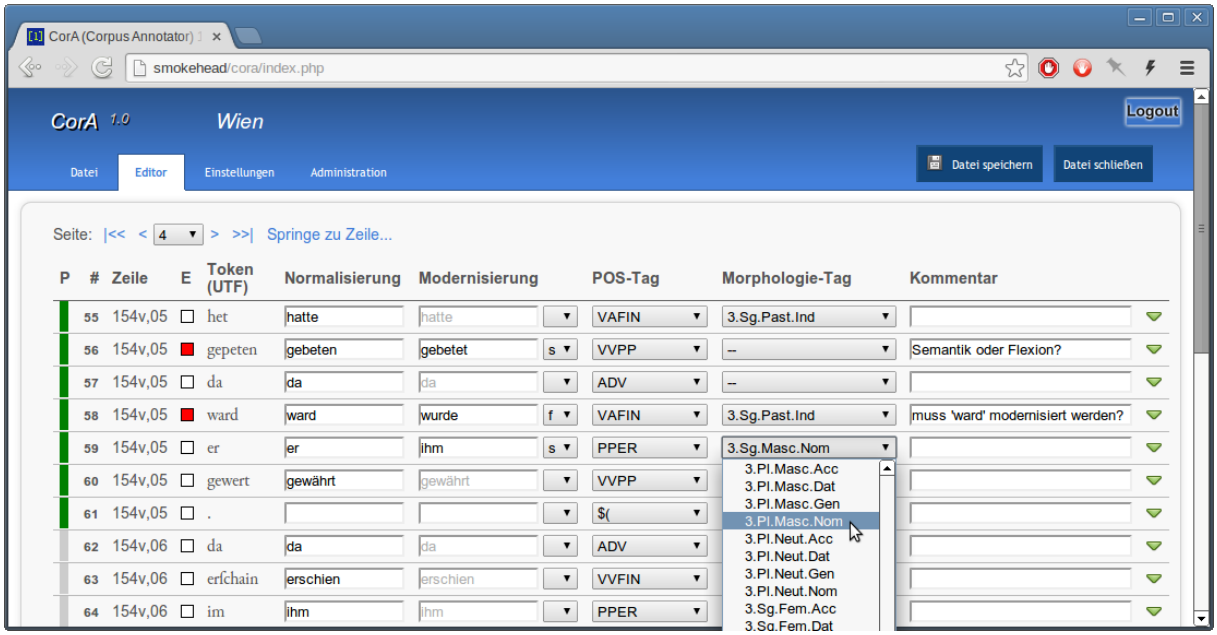

Figure 1: Web interface of CorA showing the annotation editor

of-speech tags, morphology tags, lemmatization, and (spelling) normalization. The tool is designed to increase productivity for these particular annotation tasks, while sacrificing some amount of flexibility (e.g., using different annotation layers, or annotating spans of tokens). Note that this is mainly a restriction of the web interface; the underlying database structure is much more flexible (cf. Sec. 3), facilitating the later addition of other types of annotation, if desired.

Tokens are displayed vertically, i.e., one token per line. This way, the annotations also line up vertically and are always within view. Additionally, a horizontal text preview can be displayed at the bottom of the screen, which makes it easier to read a continuous text passage. Fig. 1 shows a sample screenshot of the editor window.[3] Users can customize the editor, e.g. by hiding selected columns.

Parts-of-speech and morphology Within the editor, both POS and morphology tags can be selected from a dropdown box, which has the advantage of allowing both mouse-based and faster keyboard-based input. Tagsets can be defined individually for each text. If morphology tags are used, the selection of tags in the dropdown box is restricted by the chosen POS tag.

[3]The user interface is only available in German at the time of writing, but an English version is planned.

Lemmatization Lemma forms are entered into a text field, which can optionally be linked to a pre-defined lexicon from which it retrieves autocompletion suggestions. Furthermore, if an identical token has already been annotated with a lemma form elsewhere within the same project, that lemma is always displayed as a highlighted suggestion.

Normalization For corpora of non-standard language varieties, spelling normalization is often found as an annotation layer, see, e.g., Scheible et al. (2011) for historical data and Reznicek et al. (2013) for learner data.

In addition to normalization, an optional modernization layer can be used that defaults to the content of the normalization field. The normalization layer can be used for standardizing spelling, and the modernization layer for standardizing inflection and semantics (Bollmann et al., 2012).

Meta information CorA features a progress indicator which can be used to mark annotations as verified (see the green bar in Fig. 1). Besides serving as a visual aid for the annotator, it is also used for the automatic annotation component (cf. Sec. 2.2). Additionally, tokens can be marked as needing further review (indicated with a red checkbox), and comments can be added.

2.2 Automatic annotation

CorA supports (semi-)automatic annotation by integrating external annotation software on the server

side. Currently, RFTagger (Schmid and Laws, 2008) and the Norma tool for automatic normalization (Bollmann, 2012) are supported, but in principle any other annotation tool can be integrated as well. The "retraining" feature collects all verified annotations from a project and feeds them to the tools' training functions. The user is then able to invoke the automatic annotation process using the newly trained parametrizations, which causes all tokens not yet marked as verified to be overwritten with the new annotations.

The retraining module is particularly relevant for non-standard language varieties where appropriate language models may not be available. The idea is that as more data is manually annotated within a corpus, the performance of automatic annotation tools increases when retrained on that data. This in turn makes it desirable to re-apply the automatic tools during the annotation process.

2.3 Editing primary data

In diplomatic transcriptions of historical manuscripts, the transcripts reproduce the manuscripts in the most accurate way, by encoding all relevant details of special graphemes and diacritics, and also preserving layout information. Transcribers often use ASCII-based encodings for special characters, e.g., the dollar sign $ in place of a long s ('ſ').

The data model of CorA (cf. Sec. 3) distinguishes between different types of token representations. In the annotation editor, the user can choose to display either the original transcription layer or the UTF-8 representation.

If an error in the primary data—e.g., a transcription error or wrong tokenization—is noticed during the annotation, it can be corrected directly within the editor. CorA provides functionality to edit, add, or delete existing tokens. Furthermore, external scripts can be embedded to process any changes, by checking an edited token for validity (e.g., if tokens need to conform to a certain transcription format), or generating the UTF-8 representation by interpreting special characters (e.g., mapping $ to ſ).

2.4 Comparison to related tools

There is a range of annotation tools that can be used for enriching data with different kinds of annotations. Prominent examples are GATE, EXMARaLDA, MMAX2, brat, and WebAnno.[4] Many annotation projects nowadays require distributed collaborative working of multiple parties. The currently preferred solution is to use a tool with an underlying database which is operated through a standard web-browser. Among the tools above, only brat and WebAnno are web-based tools. Compared to CorA, these tools are more flexible in that they support more annotation layers and more complex (e.g., multi-word) annotations. WebAnno, in addition, offers facilities for measuring inter-annotator agreement and data curation. However, brat and WebAnno do not allow edits to the source document from within the tool, which is particularly relevant for non-standard language varieties. Similarly, they do not support retraining on newly annotated data.

3 Data Model

The requirements described in Sec. 2 present various challenges to the data storage, which necessitated the development of our own data model. A data model in this context is a conceptual model of the data structure that allows serialization into various representations such as XML or databases. Such a model also allows for easy conversion between serializations and hence facilitates interoperability with existing formats and tools. The complex, multi-layered layout, the differences in tokenization, and the fine-grained description of graphematic pecularities in the primary data cannot be captured well using existing formats. For example, tokenization differences as they are handled by formats such as <tiger2/> (Bosch et al., 2012) pertain only to the contraction of underlying units to original forms, and not the other way around. This means that while a conversion in such formats is easily possible, some of the data structure that is captured by our model is necessarily lost in the process. To come up with a data model that minimizes redundancy and allows for flexibility and extensibility, and accomodates the work flow of our transcriptors and annotators, we employed normalization techniques from database development. A slightly simplified version of the data model is shown in Fig. 2.

[4] GATE: `http://gate.ac.uk/`
EXMARaLDA: `http://www.exmaralda.org/`
MMAX2: `http://mmax2.sourceforge.net/`
brat: `http://brat.nlplab.org/`
WebAnno: `https://code.google.com/p/webanno/`

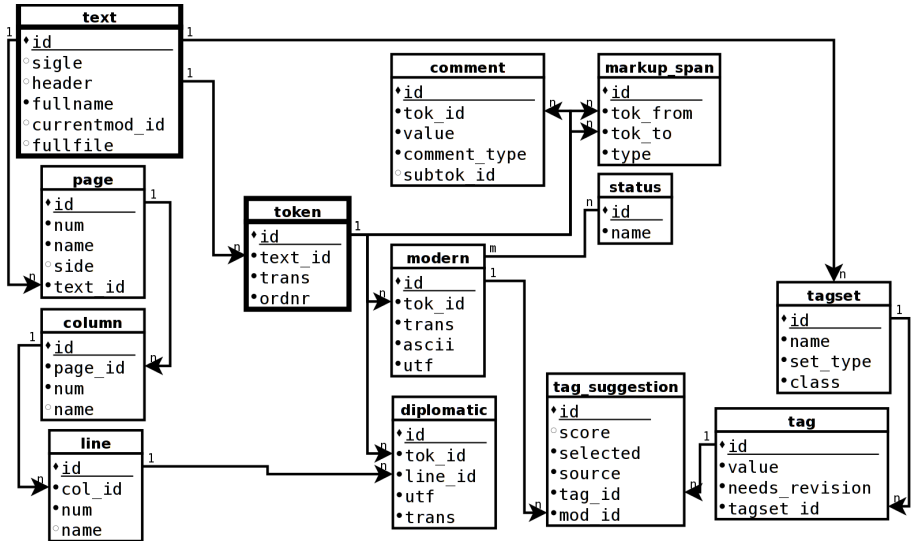

Figure 2: Data model used for CorA

Token and Text The model is centered around two units, a text and a token. A token is a virtual unit that can manifest in two ways, the diplomatic token and the modern token, each of which has a one-to-many relation with a token (cf. Fig. 3). Diplomatic tokens are tokens as they appear in the original, historical text, while modern tokens mirror modern conventions for token boundaries, representing suitable units for further annotations, e.g. with POS tags. All physical layout information on the other hand relates to the diplomatic token.

The text is the entirety of a transcribed document that can be partitioned in various ways. The layout is captured by its relation to the page, column, and line, which in turn relate to the diplomatic tokens. Furthermore, a text can be assigned one or more tagsets. The tagsets in turn can be open, such as lemmatization tags, or closed, such as POS tags. Each text can be assigned different tagsets.

Extensions In addition, the data model also allows for the import of markup annotations with the texts, which may denote layout-related or linguistic peculiarities encoded by the transcriptors, as well as information about its annotation status such as progress, or dubious annotations. The model is easily extendable for user management that can tie in to the text table, e.g., a user can be set as owner or creator of a text.

As XML serialization is not optimized for data which is not strictly hierarchically structured, storage and retrieval is rather inefficient, and extensions are not easily possible. For this reason, we chose to implement the application with an SQL database

```
<token>
    <!-- diplomatic tokenization -->
    <dipl trans="ober"/>
    <dipl trans="czugemich"/>

    <!-- modern tokenization -->
    <mod trans="oberczuge">
        <norm tag="überzeuge"/>
        <pos tag="VVIMP.Sg"/>
    </mod>
    <mod trans="mich">
        <norm tag="mich"/>
        <pos tag="PPER.1.Sg.*.Acc"/>
    </mod>
</token>
```

Figure 3: Example serialization of *ober czugemich* (modern *überzeuge mich* ‘convince me’) in XML

serialization of the data model.

4 Conclusion

We described CorA, a web-based annotation tool. Its main features are the integration of automatic annotation software, the possibility of making edits to the source document, and the conceptual distinction between diplomatic and modern tokens in the data model. We believe that these features are particularly useful for annotators of non-standard language data such as historical texts, and set CorA apart from other existing annotation tools.

We plan to make the tool available under an open source license eventually. However, we are currently still working on implementing additional functionality. In future work, we plan to integrate features to evaluate annotation quality, such as automatically calculating inter-annotator agreement.

References

Marcel Bollmann, Stefanie Dipper, Julia Krasselt, and Florian Petran. 2012. Manual and semi-automatic normalization of historical spelling – case studies from Early New High German. In *Proceedings of the First International Workshop on Language Technology for Historical Text(s) (LThist2012)*, Vienna, Austria.

Marcel Bollmann. 2012. (Semi-)automatic normalization of historical texts using distance measures and the Norma tool. In *Proceedings of the Second Workshop on Annotation of Corpora for Research in the Humanities (ACRH-2)*, Lisbon, Portugal.

Sonja Bosch, Key-Sun Choi, Éric de la Clergerie, Alex Chengyu Fang, Gertrud Faaß, Kiyong Lee, Antonio Pareja-Lora, Laurent Romary, Andreas Witt, Amir Zeldes, and Florian Zipser. 2012. <tiger2/> as a standardised serialisation for ISO 24615. In *Proceedings of the 11th Workshop on Treebanks and Linguistic Theory (TLT)*, Lisbon, Portugal.

Jennifer Foster, Ozlem Cetinoglu, Joachim Wagner, Joseph Le Roux, Stephen Hogan, Joakim Nivre, Deirdre Hogan, and Josef van Genabith. 2011. #hardtoparse: POS tagging and parsing the twitterverse. In *Proceedings of AAAI-11 Workshop on Analysing Microtext*, San Francisco, CA.

Eugenie Giesbrecht and Stefan Evert. 2009. Part-of-speech tagging — a solved task? An evaluation of POS taggers for the German Web as Corpus. In *Proceedings of the 5th Web as Corpus Workshop (WAC5)*, pages 27–35, San Sebastian, Spain.

Paul Rayson, Dawn Archer, Alistair Baron, Jonathan Culpeper, and Nicholas Smith. 2007. Tagging the Bard: Evaluating the accuracy of a modern POS tagger on Early Modern English corpora. In *Proceedings of Corpus Linguistics 2007*, University of Birmingham, UK.

Marc Reznicek, Anke Lüdeling, and Hagen Hirschmann. 2013. Competing target hypotheses in the Falko Corpus: A flexible multi-layer corpus architecture. In Ana Díaz-Negrillo, Nicolas Ballier, and Paul Thompson, editors, *Automatic Treatment and Analysis of Learner Corpus Data*, pages 101–123. Amsterdam: Benjamins.

Silke Scheible, Richard J. Whitt, Martin Durrell, and Paul Bennett. 2011. Evaluating an 'off-the-shelf' POS-tagger on Early Modern German text. In *Proceedings of the ACL-HLT 2011 Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH 2011)*, pages 19–23, Portland, Oregon, USA.

Helmut Schmid and Florian Laws. 2008. Estimation of conditional probabilities with decision trees and an application to fine-grained POS tagging. In *Proceedings of COLING '08*, Manchester, Great Britain.

Amir Zeldes, Julia Ritz, Anke Lüdeling, and Christian Chiarcos. 2009. ANNIS: a search tool for multi-layer annotated corpora. In *Proceedings of Corpus Linguistics*, Liverpool, UK.