A Flexible NLP Pipeline for Computational Narratology

Thomas Bögel **Jannik Strötgen** **Christoph Mayer** **Michael Gertz**
Institute of Computer Science, Heidelberg University, Germany
{boegel, stroetgen, cmayer, gertz}@uni-hd.de

1 Project Overview

Temporal dependencies reveal interesting insights into the semantic discourse structure of narrative texts. The investigations of literary scientists are, as of today, mostly based on labor-intensive manual annotations. Computational Narratology, an important subtopic of the Digital Humanities, aims at facilitating annotations and supporting literary scientists with their analyses. According to Mani (2013), one aspect of Computational Narratology focuses on exploring and testing literary hypotheses through mining narrative structures from corpora. In the context of the BMBF-funded eHumanities project **heureCLÉA**, we address temporal phenomena in literary text, a genre whose temporal phenomena are different from others. For example, it is often not possible to anchor temporal expressions to real points in time, but literary texts tend to have their own time frame. Our project partners, as well as many other humanists, use CATMA, a comprehensive graphical tool for annotating data. Interfacing NLP with CATMA could drastically reduce the effort of manual annotation. The goal of heureCLÉAis to provide users with a collaborative annotation environment for tagging temporal phenomena in documents, with simple annotations (e.g., temporal expressions) being added automatically, and more complex annotations (e.g., time shifts and ellipses) being suggested. Users can correct automatic annotations, and user feedback will be used to apply machine learning techniques to improve future annotation suggestions.

In the following, we outline our flexible architecture for NLP in the domain of narrative texts, as well as promising first results for annotating the tense of sub-sentences to demonstrate the effectiveness of our approach.

2 Architecture and Components

The heureCLÉAcorpus currently consists of more than 20 mostly German narrative texts from various authors of the 20th century. Due to the diversity of style and text characteristics, applying NLP is challenging as most systems are optimized for factual texts characterized by stable structures.

Automatically generating annotations that are related to temporal structures of texts requires information on multiple levels of the linguistic processing stack. Thus, we implemented a modular pipeline that performs annotations with increasing levels of complexity and allows for easy adaptation and exchange of different base components. We use standard off-the-shelf tools that are freely available. In order to achieve maximum flexibility and to allow for easily substitutable individual components, the pipeline is implemented as a UIMA architecture. The general pipeline architecture is shown in Fig. 1. We distinguish between general preprocessing components that are required for all subsequent narratological annotation tasks and individual machine learning modules (red background) that are tailored to one specific target annotation. We are planning to release the preprocessing stack to the research community to allow all CATMA users to perform basic NLP analyses and annotations.

2.1 Component Overview

2.1.1 CATMA Interface

The texts in our corpus are annotated by literary scientists with CATMA, a web-based collaborative

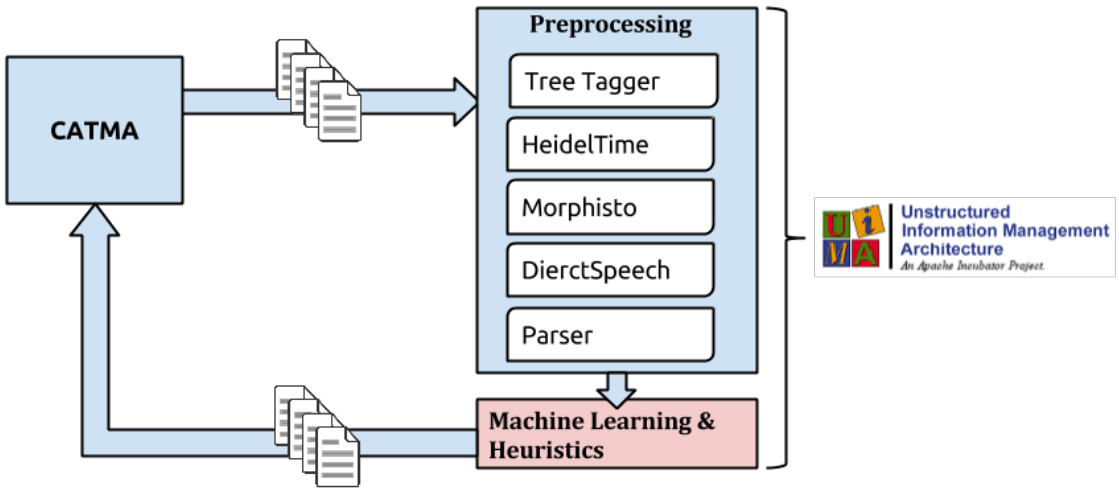

Figure 1: Overview of the NLP architecture in heureCLÉA.

annotation tool that offers humanists an easy way to create stand-off annotation and share their annotation with other scholars. In order to work with annotated data in CATMA, we implemented a component that interfaces CATMA with our UIMA pipeline. As CATMA is a popular tool in the humanities, we developed the interface as a stand-alone component that can easily be used by others to combine the strengths of CATMA as an annotation framework with the analytical and predictive power of UIMA pipelines. The interface is geared to literary scientists with no knowledge of programming. To achieve a simple configuration, the user only has to specify mappings between CATMA and UIMA types in a single XML file. By providing an easy-to-use interface, we want to lower the bar for other projects in the Humanities to employ simple yet effective NLP tools and thereby alleviate manual annotations.

2.1.2 Linguistic Processing

Our linguistic preprocessing stack consists of state-of-the-art components for German NLP. For sentence segmentation and part-of-speech tagging, we use the TreeTagger (Schmid, 1995). While the tree tagger provides a basic analysis of tokens, it does not extract morphological information; thus, we added Morphisto (Piskorski, 2009) as a separate component for morphological analysis. Two different parsers in our pipeline provide valuable information for detecting sub-sentences and dependency relations between tokens (e.g., to extract the subject of a certain verb): the Stanford constituent parser (Rafferty and Manning, 2008) and ParZu (Sennrich et al., 2009), a dependency parser trained on the TüBa-D/Z. Finally, HeidelTime (Strötgen and Gertz, 2013) extracts and normalizes temporal expressions in the text which will be used in later stages of the processing pipeline.

2.1.3 Machine Learning

After the data has been annotated by the above preprocessing components, it is passed to different modules that handle the annotation of specific narratological aspects of texts (e.g., the extraction of tense clusters of time shifts). Depending on the target annotation, we employ different machine learning approaches and heuristics.

Verb tenses are an example for such a relevant annotation because shifts in the verb tense of a sentence can, for instance, indicate narratological order phenomena (e.g., prolepsis). To extract tense clusters, we implemented and evaluated a robust heuristic component with promising results. Detailed evaluation results of the prediction performance based on a comparison to manually annotated data will be presented in the poster.

References

Mani, I. (2013, October). Computational narratology. the living handbook of narratology. http://www.lhn.uni-hamburg.de/article/computational-narratology.

Piskorski, J. (2009). Morphisto-An Open Source Morphological Analyzer for German. *Finite-state Methods and Natural Language Processing: Post-proceedings of the 7th International Workshop FSMNLP 191*.

Rafferty, A. N. and C. D. Manning (2008). Parsing three german treebanks: Lexicalized and unlexicalized baselines. In *Proceedings of the Workshop on Parsing German*, PaGe '08, Stroudsburg, PA, USA, pp. 40–46. Association for Computational Linguistics.

Schmid, H. (1995). Improvements in part-of-speech tagging with an application to german. In *In Proceedings of the ACL SIGDAT-Workshop*, pp. 47–50.

Sennrich, R., G. Schneider, M. Volk, and M. Warin (2009). A new hybrid dependency parser for german. *Proc. of the German Society for Computational Linguistics and Language Technology*, 115–124.

Strötgen, J. and M. Gertz (2013). Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation 47*(2), 269—-298.