A Tool for Error Detection in Automatically Annotated Latin Text

Julius Steen
Universität Heidelberg
steen@cl.uni-heidelberg.de

Ines Rehbein
Leibniz ScienceCampus
rehbein@cl.uni-heidelberg.de

Josef Ruppenhofer
Leibniz ScienceCampus
ruppenhofer@ids-mannheim.de

Tag
Datum
Zeit
Raum

Many projects in the Digital Humanities rely on automatic NLP tools to annotate their data as a basis for further research. The quality of the automatic annotations, however, is not always good enough while human post-processing is often too costly and thus not available to most projects. What we need is a cheap and yet efficient way to detect errors in automatically annotated text so that they can be corrected by human annotators without proof-reading the whole corpus.

We present such a method and test it on the task of finding parts-of-speech errors in automatically annotated medieval Latin text. The data we use comes from the Corpus Thomisticum (Bamman et al. 2007) which contains texts by Thomas Aquinas and contemporary authors, manually tagged for POS. We first train 7 freely available POS taggers on the data to obtain the automatic predictions. The tagging accuracy of the individual taggers ranges between 92.4-96.8%, with an average accuracy of 94.8%. Then we use a Bayesian Inference model to compute the posterior entropies for the taggers' predictions, which can be interpreted as the confidence of the model in the predictions (Hovy et al. 2007). We use these scores to select instances in an active learning framework and collect human judgements for the selected instances. These judgements are then fed back to the model, and the parameters are updated (Rehbein and Ruppenhofer 2017).

We show that using feedback from active learning to guide the Bayesian Inference model results in high precision and recall for error detection in automatic annotations, thus reducing the time and effort needed for manual error correction, which makes the tool a valuable assistant for DH projects. Our approach is not restricted to POS annotations but can also be applied to other classification tasks, such as Named Entity Recognition.

References: • Bamman et al. (2007): *Guidelines for the syntactic annotation of Latin treebanks* (v1.3). Tech.report. Tufts University Digital Library. • Rehbein and Ruppenhofer (2017): Detecting annotation noise in automatically labelled data. *Proceedings of ACL 2017*, 1160–1170. • Hovy et al. (2013): Learning whom to trust with MACE. *Proceedings of HLT-NAACL 2013*, 1120–1130.