RUPRECHT-KARLS-
UNIVERSITÄT HEIDELBERG
ZUKUNFT SEIT 1386

Institut für Computerlinguistik Heidelberg

Seminararbeit im Seminar
Language Technology for Education Assessment
Wintersemester 2017/2018

Addendum to: Language Level Analysis and Classification

Referenten:
Dr. Magdalena Wolska & Dr. Ines Rehbein

Verfasserin:
Julia Suter
Matrikelnummer 3348630
Masterstudiengang Computerlinguistik
6. Fachsemester

30. September 2018

1 Introduction

The original work was aimed at analyzing and classifying language levels. However, the same methods may be applied to any corpus and any classification task. As a proof of concept, I extracted the very same features from a collection of literary texts by different authors and used them to train an author classifier.

2 Dataset

The collection of German text samples was extracted from Project Gutenberg. It contains text samples from 20 different authors. As it was done in the original work, the texts were parsed by ParZu and CorZu and thereafter split into chunks of 50 sentences, yielding 7365 samples. At least 100 samples representing at least 10 different works are available for each author. For each sample, the 80 language level features described in the original work were computed.

3 Experiments

I repeated the core experiments from the original work to investigate the differences in performance on this new text collection. Except for the fact that here 80% of the samples are used as training data (previously 90% due to sample shortage) the settings of the experiments were kept identical. Note that no parameter screening has been conducted for these experiments. Table 1 shows the results for the baseline system, the default linear SVC with 80 features and 3 systems with reduced feature sets using feature agglomeration.

3.1 Baseline and Default System

As expected, the baseline system with 2 features performs worst with 23.5% accuracy. Note that the 2 features are meant to measure readability so they are less suitable for author classification than for language level classification. Nevertheless, the baseline clearly outperforms a guessing approach, which would reach roughly 5% for stratified experiments with 20 classes.

The Linear SVC with 80 features reaches almost 97.9% accuracy. Given the number of classes and relatively simple training algorithm, this performance is surprisingly high. Possible explanations for why classification works better on this dataset include the larger number of samples, the longer text length and the less pronounced feature sparsity (as discussed below).

3.2 Feature Coefficients and Relevance

Figure 1 shows the SVC coefficient strength for each feature on each author class. When comparing the 15 strongest coefficient features with the ones from the previous experi-

	# Features	Accuracy ± SD	Precision	Recall	F1
Baseline	2	0.235 ± 0.020	0.121	0.235	0.137
Linear SVC	80	0.979 ± 0.004	0.979	0.979	0.979
Feat. aggl. (sparse f.)	65	**0.983** ± 0.003	**0.984**	**0.983**	**0.983**
Rel. features only	15	0.952 ± 0.007	0.955	0.952	0.952
Feat. aggl. (less rel. f.)	20	0.959 ± 0.007	0.961	0.959	0.959

Table 1: Author classification for baseline, default and feature agglomerated systems.

ments on language level classification, one can observe an overlap of 11 features: *modal, consecutive, final, interrogative, causal, other, concessive, temporal, conditional, nouns,* and *A2 features summed*. 9 of them are clause features, suggesting that clause features are exceptionally suitable for text classification tasks, even on different types of texts. Modal clauses, for instance, are highly indicative for texts by Bierbaum, Schwab and Fontane, while they are less likely to appear in works by Zweig and Thoma.

However, there are some features that receive a different ranking in this new experiment set: *punctuation marks* and *discourse entities*, for instance, receive high coefficients while they only show moderately high coefficients in the language level classification task. Applying this method on more text collections and classification tasks may reveal which features generally receive high coefficients and which are especially relevant for specific domains or classification tasks.

3.3 Feature Agglomeration

As in the original work, I experimented with feature agglomeration to improve the results from the default system using all 80 features. I used the same two methods to identify sparse or less relevant features to be agglomerated.

I considered all features that have a 0 value in more than one third of the samples as sparse. While this method revealed more than half the features as sparse in the original experiments, it identifies only 20 sparse features for this dataset. The reason for this reduced feature sparsity may be that most authors write at a high language level that includes many of the structures and linguistic phenomena captured by the features, while language learning texts are usually written in a more simple fashion. Furthermore, the literary work samples are considerably longer than the ones in the language level dataset, which increases the probability for a linguistic feature to occur. When clustering the 20 sparse features into 5 features and adding them to the 60 non-sparse features, classification performance is slightly improved, reaching an accuracy of 98.3%, although this change is not statistically significant.

I also repeated the feature agglomeration experiments with the less relevant features, which are identified by their low SVC coefficients rank. In the original experiments, considering only the 15 most relevant features improves performance by almost 7%. Here,

however, we observe a decrease in performance of roughly 2% in accuracy. Similarly, using only the 15 most relevant features and clustering the remaining into 5 features decreases performance. In the original work, these two methods yielded the highest results. The low performance here indicates that the less relevant features are not simply noise in this dataset but contain relevant author information not represented by the 15 most relevant features.

4 Conclusion

The experiments showed that the 80 features originally designed for language level classification can successfully be applied to other domains, text types and classification tasks. They have proven to be general features for capturing syntactic differences in texts.

The feature sparsity problem discussed in the original work does not apply as strongly to this dataset. Nevertheless, feature agglomeration on sparse features slightly, albeit not significantly improves performance. The feature relevance measured by SVC coefficient does not seem to be as suitable in identifying the strongest features as it was in the original work. Feature agglomeration on less relevant features did not improve performance, which shows that feature agglomeration is not suitable (or necessary) for all kinds of datasets. However, it remains a valid option for improving the default system in certain tasks, depending on the distribution of linguistic phenomena and degree of feature sparsity.

In future work, one could investigate in detail which features are reliable indicators for specific authors. The features may reveal in what way authors' writing styles differ from each other or how they changed over time. Clustering authors or literary works using these features could help identify texts with similar writing styles, also for unlabeled samples. Furthermore, the features may be applied to other corpora and classification tasks, for example to genre classification. Finally, the feature set, although shown to be general already, may be extended and refined to capture even more linguistic phenomena and writing style characteristics.

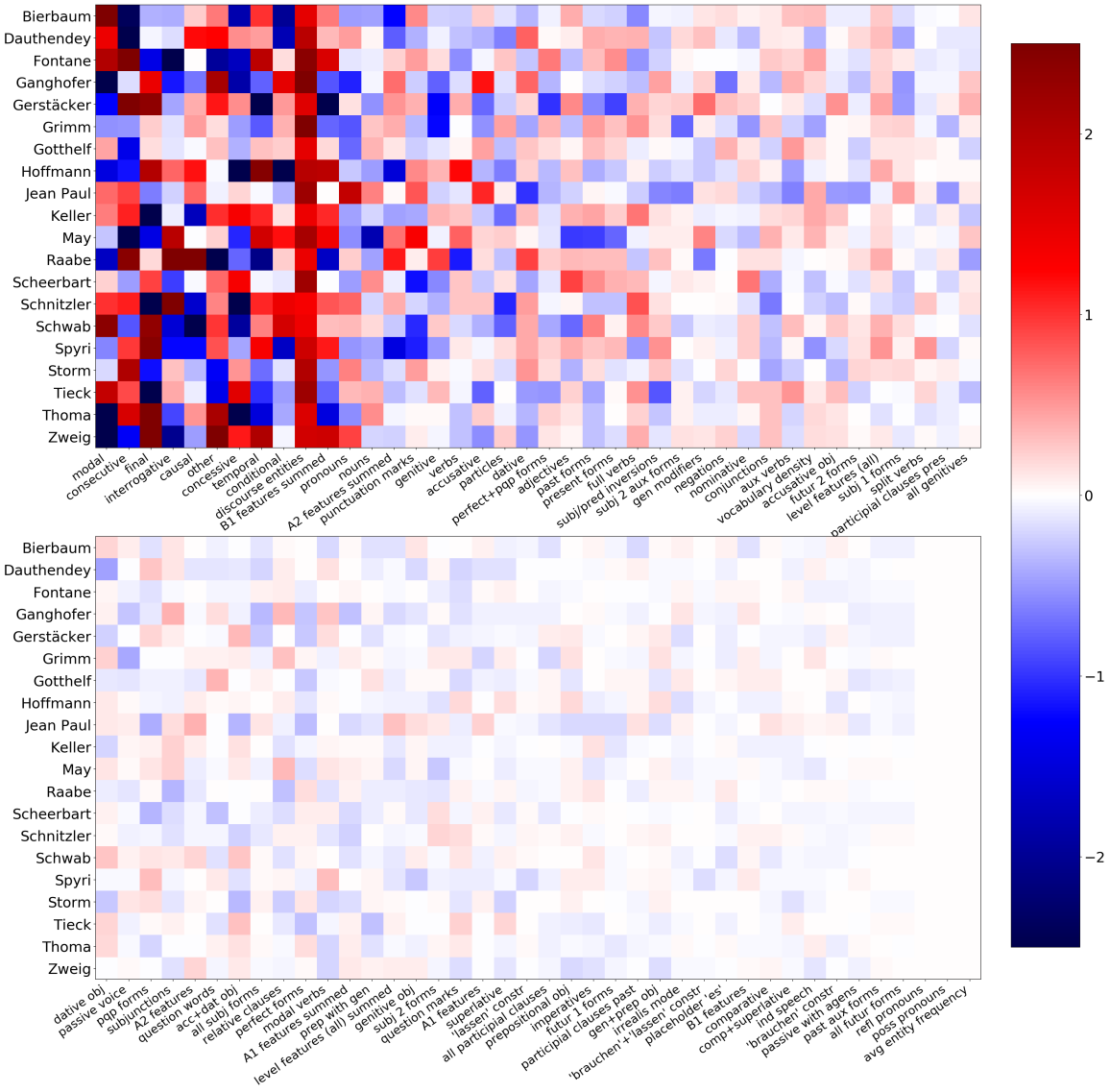

Figure 1: Features sorted by coefficient strength.