Named Entity Recognition in the *Regesta Imperii* - Person and Location Names in the late Middle Ages

Jana Hänßler
Technische Universität Darmstadt
Institut für Sprach- und Literaturwissenschaft
Computer Applications in Linguistics
Dr. Sabine Bartsch

Abstract

Recognizing entities in texts can be done automatically. In the study at hand, a model for Named Entity Recognition in a special corpus, the Regesta Imperii, is trained. The performance evaluation of the given tags for person names, titles, and location names reveals weaknesses which are probably caused by an insufficient amount of training data. Nevertheless, an analysis of names in the late Middle Ages is made by means of the tagger's results.

1 Introduction

Jurafsky and Martin (Jurafsky and Martin, 2017) define Named Entity Recognition (NER) as the task of "[...] finding spans of text that constitute proper names and then classifying the entities being referred to according to their type." A more precise definition of NER is the one of Bartsch and Weber (Bartsch and Weber, 2015). They give an overview over the purpose and the usage of NER:

> The detection of named entities can be useful for computational analysis of literary texts. Questions concerning the narrative structure can be answered: Which characters are introduced in which part of the text, which places are mentioned? Moreover, named entity recognition is a subtask in the field of information extraction. It is an important component if you are dealing with question answering or co-reference resolution.6

The topic of and the interest in NER is more up to date than ever before. As the quantity of (digital) texts increases massively in the present age, science and economics seek to analyze these masses of text. In a previous study[1], two tools for the task of NER the *Stanford Named Entity Recognizer* (SNER) and the application *ANNIE* with a German gazetteer in *GATE* were used and tested. As a result of this study, neither the one nor the other worked satisfyingly. For this reason, this paper presents the training of a named entity recognizer, which, in the end, is supposed to yield better results than the ready-made ones. The aim of this study is to do NER for person and location names in the *Regesta Imperii*[2] (RI). If the tagger does not yield better results than those from the previous study, suggestions for its improvement shall be derived from the study at hand. The idea is, however, to test borders of feasibility of the SNER instead of programming a perfectly working NE-tagger.

One kind of purpose of NER and of the research presented in this paper (and in general, too,) is to contribute to a faster recognition of names in a big corpus and therefore to a better understanding as well as further processing of the data (in the RI project). Entities here person and location names in the RI shall automatically be recognized and tagged. The basic idea here is that these automatically tagged entities can then be examined and further processed. From a linguistic point of view, the structure of the found entities, especially of the person names, is interesting. What do names in the RI look like? How are they represented in this special corpus? Furthermore, the search function of the *RI online* and accordingly the data provision for the scientific user may be improved by the present work.

After presenting the corpus and its challenging

[1] For further details the interested reader may see Hänßler (2016) *Hausarbeit NER theoretisch* a paper which introduces and compares these two types of NER with one another.

[2] The project *Regesta Imperii* will be introduced in chapter 3 The Corpus.

characteristics in chapter 2, chapter 3 depicts how the training of a NER-model was carried out, what is needed for its accomplishment, and gives short annotation guidelines. The last chapter attends to the concrete NER task and its evaluation. Furthermore, chapter 4 takes a closer historic-cultural look at the recognized person names and their linguistic appearance.

2 The Corpus

2.1 The Project Regesta Imperii

The corpus being under examination in this study is the texts of the project *Regesta Imperii* (RI). It is a long-term project of three academies; the *Academy of Sciences and Literature — Mainz*, the *Austrian Academy of Sciences* and the *Berlin-Brandenburg Academy of Sciences and Humanities*. In 1829, the Frankfurt city librarian Johann Friedrich Böhmer began with the collection of imperial documents, which actually planned as groundwork for the *Monumenta Germaniae Historica* soon developed into an independent fundamental work (cf. Webseite der Regesta Imperii, Akademie der Wissenschaften und der Literatur; henceforce referred to as RIAdW; /unternehmen[3]). The whole RI collection of regestas is subdivided into 14 sections, from Carolingians to Maximilan I. (ca. 751-1519). "The REGESTA IMPERII (RI) chronologically record all activities evidenced by charters and other sources of the Roman-German kings and emperors from the Carolingians up to Maximilian I (ca. 751-1519) as well as of the propes of the early Middle Ages and High Middle Ages in form of German Regesten (abstracts)." (RIAdW[4])

Regestas, in general, are short texts about documents; the RI regestas are short texts about medieval emperor documents. "A regesta sums up the essential content of a document in modern German and, moreover, offers statements about transmission and research state." (RIAdW[5]) The structure of each regesta is its text, the transmission and commentary as well as date and place of issue (RIAdW[6]). Currently (effective April 14, 2018), there are 90 printed volumes and 182.315 regesta online available as full text records via REST API (RIAdW[7]) in the RI Database which is licensed under a CC-BY 4.0 International[8] license.

2.2 Challenges of the Corpus

Bartsch and Weber (Bartsch and Weber, 2015) define entities as "[...] elements having a distinct, separate existence. These phrases contain, for instance, the names of persons, organizations, locations, times and quantities." From the viewpoint of linguistics, the regesta texts and its entities are interesting, but they also pose several challenges on the NE tagger.

As a first challenge, Benikova et al. (Benikova et al., 2014) state that German is difficult for NER (in comparison to English e.g.) because not only proper names are capitalized but also nouns. The RI are mostly written in German language. Some regesta texts may contain Latin or French passages. Secondly, many abbreviations in the corpus[9] may pose a difficulty for the NER process. *Knig Friedrich*, for instance, usually is abbreviated as *Kg. F.*, many other titles, too. Especially since the CRF classifier uses the word sequences in order to learn how to tag entities, this might problematic. Usually, person and location names can be differentiated and extracted by NER systems. However, as the previous study and the examination of the *Kumuliertes Register der bislang erschienenen Hefte. Regesten Kaiser Friedrichs III. (1440-1493)* (Rübsamen and Manz, 2013) of all names and locations reveals, indications of person names often come along with a location name. In other words, many person names overlap with a location name. This is aggravated by the fact that titles, too, overlap with person names.

3 The Tool

3.1 Training a Model

The tool used in this study - or rather a component of it, namely its classifier - is the *Stanford Named Entity Recognizer*. Because the pre-trained SNER-model did not yield satisfying results as shown in the previous study, it was the aim of the study at

[3] http://www.regesta-imperii.de/unternehmen.html.

[4] http://www.regesta-imperii.de/en/research.html.

[5] http://www.regesta-imperii.de/unternehmen/videopraesentation.html.

[6] http://www.regesta-imperii.de/unternehmen/ri-online.html.

[7] http://www.regesta-imperii.de/en/daten.html.

[8] https://creativecommons.org/licenses/by/4.0/.

[9] Only the summary of the regesta document content is relevant.

hand to improve the SNER-model or rather to train an own model for this special corpus.

For the training of an own model, Stanford offers a vague documentation[10]. The process basically is a triad. In a first step, training data needs to be built. This "training data needs to be in tab-separated columns, with minimally the word tokens in one column and the class labels in another column." To give a limit to the number of texts worked with, a fraction of one volume - *[RI XIII] H. 14 - Friedrich III., Nürnberg 1 (1440-1449)* - was chosen for this project; 21 of the entries (3115 tokens) from *Heft 14 Friedrich III., Nürnberg (1440-1149)* were 1) tagged with O (default value) by a *Python* script[11] and 2) manually NE-tagged and served as training data.[12] The second step in the training process was to actually train the model. Therefore, a properties file[13] was created.

Via the *Windows PowerShell* and the command in figure 1 the training process is started.

```
PS C:\Users\janah\workspace\NER_tagger_GER> java -cp stanford-ner.jar
 edu.stanford.nlp.ie.crf.CRFClassifier -prop properties_training.prop
```

Figure 1: Command for Training

The classifier behind the SNER is a Conditional Random Field (CRF) classifier. While other models predict a tag only based on a single word, a CRF classifier also takes important pieces of information from the text - sequential data - to compute a probability of the output; it also considers the "neighboring" words. The NE tag with the highest probability is then applied. In a third step, the trained model was applied to the test data which consisted of 67 regesta texts from *Heft 14 Friedrich III., Nürnberg (1440-1149)*. This test file also had to be tokenized and labeled with the default value before applying the model to it.

```
PS C:\Users\janah\workspace\NER_tagger_GER> java -cp stanford-ner.jar
 edu.stanford.nlp.ie.crf.CRFClassifier -loadClassifier ner-model-test
-IOB.ser.gz -testFile test_tokens_from_trainingData.txt > output_TIT_
lo.txt
```

Figure 2: Command for Testing

The tagged data was outputted to the folder in which the model was saved.

PERS	TIT	LOC
Karl Holzschuher	*Kg. / Gff. / Hz. / Bf.*	*Nürnberg/ Pappenheim*
Kg. F.	*Erbmarschall, Fürsten, Ritter*	*Altmühl*
Erbmarschall und Ritter Konrad von Pappenheim	*Bürger-meistern, Rat und Bürgern der Stadt Nürnberg*	*Weienburg zu dem Kreuz*
Pappenheimer/ Nürnberger	*Kaisern und Königen*	*heyligen römischen reiche*
Nürnberger Bürger	*Juden*	*Brücke bei Bubenheim*
Heinrich	*kgl. Kammer*	
	Richter, Amt-leute, Zöller, Mautner	
	Landgericht / Haus + LOC	

Table 1: Entity Examples

3.2 Annotation Guidelines

This section gives a short overview of the guidelines which determined the annotation of the chosen entity classes. The object of interest is names of persons and locations.[14] Therefore and because person names in the RI are more complex than prename and name, the entity classes which were annotated are *titles* (TIT), and names of *persons* (PERS) as well as *locations* (LOC). The following table, which is anything but complete, gives an impression of which words or phrases were regarded and annotated as PERSon, TITle and LOCation:

The entity class PERS contains full names as well as the abbreviation *Kg.F.*, standing for *König Friedrich*. Furthermore, beholders see that the PERS entities do not only contain names, but that they as well consist of titles (e.g. *Erbmarschall*), locations (e.g. *Nürnberger*) and other supplements (e.g. *Bürger*). The entity *Nürnberger Bürger* tagged as PERS has to be explained, especially in contrast to *Bürgermeistern, Rat und Bürgern der Stadt Nürnberg* tagged as title. Since the latter frequently occurs as an established term contain-

[10] https://nlp.stanford.edu/software/crf-faq.html#a.

[11] The script can be found in the GitHub repository https://github.com/JanaHae/CAL.

[12] The labeled training data can be found in the GitHub repository https://github.com/JanaHae/CAL.

[13] The properties file can be found in the GitHub repository https://github.com/JanaHae/CAL.

[14] Pronouns were not tagged as entities.

ing the titles *Bürgermeister* and *Rat*, it appears as TIT. As opposed to this, *Nürnberger Bürger* appears frequently in combination with a PERS name (e.g. *Nürnberger Bürger Thomas Müllner*) and consequently is tagged as such. In order to meet the requirements of recognizing these complex structures of many full names in the RI, including their titles, for instance, or other supplements, these were also tagged as PERS. Therefore, the entity classes were completed by affixes in form of the IOB[15]-schema (see table 2). This schema allows annotating phrases, as in to show dependencies and entities consisting of more than one token. The TIT-tag, in the first place, was intended to denote all the ruler and office titles. During the annotation process, however, soon a need for job titles (e.g. *Richter*), titles of institutions (e.g. *kgl. Kammer*), and titles for groups (e.g. *Juden*) emerged. As a consequence, the scope of utilization of the TIT-tag was extended. Titles are also annotated when they appeared by themselves. The LOC-tag is a feature of all locations, whether the location is the name of a town (e.g. *Nürnberg*), of a territory (e.g. *heyligen römischen reiche*), or a local entity specified by a location (e.g. *Brücke bei Bubenheim*). Location names were tagged without supplements as for instance *Stadt*.

Since registers of all RI volumes already exist, the annotation guidelines for the NER are based on these. To be more precise, the author and annotator oriented her guidelines towards the import of the regesta texts and their visualizations in the graph platform *Neo4j*[16]. This does not mean that the entity annotation and the entities given in the RI registers agree completely. Rather the annotator could apprehend the entities easier than from the registers and, besides, develop a feeling for which entities seemed to be important to the regesta writers by looking at the graph representation. In the following, some examples (in one cases in combination with a screenshot from the respective regesta in *Neo4j*) shall give an idea of which entities were tagged how and how the graph visualization could help with the understanding of the entities.

This example of an extract from a regesta text (table 2) and figure 3, a partial screenshot from the regesta texts import into *Neo4j* show how the graph representation helped the annotator verify

PERS	**TIT**	**LOC**
zwischen	O	O
Mgf	B-PERS	B-TIT
.	I-PERS	I-TIT
Friedrich	I-PERS	O
II	I-PERS	O
.	I-PERS	O
von	I-PERS	O
Brandenburg	I-PERS	B-LOC
,	I-PERS	O
Kf	I-PERS	B-TIT
.	I-PERS	I-TIT
und	I-PERS	O
hauptmann	I-PERS	B-TIT
der	I-PERS	I-TIT
sachen	I-PERS	I-TIT
,	O	O
den	O	O
Hzz	B-PERS	B-TIT
.	I-PERS	I-TIT
Johann	I-PERS	O
IV	I-PERS	O
.	I-PERS	O

Table 2: Illustration data format. Use of the IOB-schema in order to nest entities. The regesta extract contains two PERS entities with each one TIT. Additionally, the first name also contains a LOC.

[15]Internal - Other - Beginning.
[16]https://neo4j.com/.

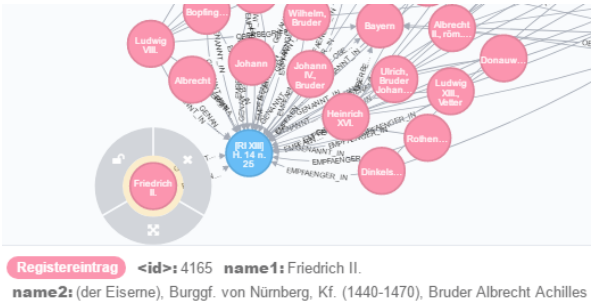

Figure 3: Screenshot of a Regest in *Neo4j*

if the interpretation of cohesive words was right. From the tokenized text it does not become clear if *Mgf. Friedrich II. von Brandenburg* is the *Kf. und hauptmann der sachen* in one person, or if they are two persons. By looking it up in the *Registereintrag* of the graph database, one can find out that *Friedrich II.* also bears the title *Kf.*. As a consequence, the tagging had to continue with *I-PERS* instead of *B-PERS* because still the same person is meant.

4 Results and Outlook

To begin with, the last chapter presents the results of the study. These can be looked at from a methodological perspective and from a historico-cultural one. To keep this paper in the provided extent, single but representational regesta texts/entities and the performance of the self-trained NE-tagger (on them) will be examined. This analysis proceeds to an outlook with remarks on how to improve the NER in the RI.

4.1 Methodologically

During the training process, the author was faced with some challenges. The next few lines describe these difficulties and how they have been resolved. Initially, the model should learn to tag all of the three entity classes. However, when looking at the output of the model, it became apparent that there must have been a problem with the three-column-format in the settings of the properties file. In order to circumvent the extra time for 1) testing many properties settings or 2) renewed annotating, the annotated training file was split into three files by a *Python* script, so that each of the three files then contained only one entity class. Subsequently, three models were trained on these files and the resulting models were run on the same test file. Afterwards, another *Python* script merged the test files into one with three columns (LOC in first column, PERS in second column, TIT in third column).

Gf	*O*	*B-PERS*	*B-TIT*
.	*O*	*I-PERS*	*I-TIT*
Ulrich	*O*	*I-PERS*	*O*

Table 3: Example 1

Nürnberger	*O*	*B-PERS*	*O*
Bürger	*O*	*I-PERS*	*B-TIT*
Konrad	*O*	*I-PERS*	*O*
Aschpach	*O*	*I-PERS*	*O*

vs.

Konrad	*O*	*O*	*O*
Aschpach	*O*	*O*	*O*

Table 4: Example 2

For this study, the train-test-split was done on the same *Regestenheft (Nr. 14)*. That means that the model was trained and tested on relatively similar regesta texts. The author is aware that the classifier thereby just could have overfitted the model (i.e. 'memorized' some terms instead of recognizing similar word sequences and tagging them on the basis of a learned text structure). When the amount of training data is enlarged for a follow-up study, the model should also be tested on - and probably work better for - texts from other parts of the RI, too.

Below, the results of the tagging with the self-trained model are analyzed. This analysis is not exhaustive, but still it gives an idea of how well the tagger functions, and where there lie its weaknesses.

The name *Ulrich* does not occur in the training data. Therefore, it is safe to assume that the model has learned the structure - that a title is followed by a name and that this, as a whole, is a person (see table 3). Another example to confirm this, is the case that is shown in table 4: While the name *Konrad Paumgartner* with the supplement *Nürnberger Bürger* is recognized as a PERS entity, *Konrad Aschpach*, without such a supplement, is not recognized.

Here, however, a weakness of the tagger is revealed. In the current version of the NER, names without supplements are hardly recognized. A first glance into the cumulated register of the RI raised an assumption about the above-mentioned chal-

Mgf	*O*	*B-PERS*	*B-TIT*
.	*O*	*I-PERS*	*I-TIT*
Albrecht	*O*	*I-PERS*	*O*
von	*O*	*I-PERS*	*O*
Brandenburg	*B-LOC*	*I-PERS*	*O*

Table 5: Example 3

Mgf	*O*	*B-PERS*	*B-TIT*
.	*O*	*I-PERS*	*I-TIT*
Johann	*O*	*I-PERS*	*O*
von	*O*	*I-PERS*	*O*
Brandenburg	*B-LOC*	*I-PERS*	*O*
aufgrund	*O*	*I-PERS*	*O*
der	*O*	*I-PERS*	*O*
Appellation	*O*	*I-PERS*	*O*
Georg	*O*	*I-PERS*	*O*
Geuders	*O*	*I-PERS*	*O*
von	*O*	*I-PERS*	*O*
Nürnberg	*B-LOC*	*I-PERS*	*O*

Table 6: Example 4

lenge for an entity recognizer: that the nesting of the entity classes would cause problems. In many examined regesta texts, people carry the name of a location (and sometimes even a title, too,) in their personal name, as table 5 shows.

In this case as in many other with the same structure, the nesting did not cause a problem. Both the PERS-tag and the LOC-tag as well as the TIT-tag were correctly applied. Differently, the tagger discovered some difficulties with the PERS-tag in the following phrase that is shown in table 6.

Probably because the words in between the two entities *aufgrund der Appellation* are not known by the tagger, it continued with the I-PERS-tag until a known ending or rather until a word which it did not recognize as an entity. The next example shows that the tagger can differentiate if it should annotate with a PERS-tag and a TIT-tag or only with a TIT-tag. While *Kg.* sometimes appears without a proper name and thus should only be tagged with TIT, *Kg. F.* can be recognized as PERS as table 7 shows.

Another weakness is demonstrated in the comparison of the next two examples. In the first of these two, *Gericht* is not labeled although it should be tagged as (B-)TIT because it appears with a proper town name. As a consequence, *Heroldsberg* should not only be tagged as LOC but also as I-TIT in order to show the cohesiveness. Job titles should also be tagged as TIT. While the NER knew to annotate *Landrichter* as such, it failed to label *Rechtsprecher*. The same holds true for the town *Sulzbach* in contrast to *Heroldsberg*. Additionally, this example also shows the limits of this kind of tag schema. *Landrichter* und *Rechtsprecher des Landgerichts Sulzbach* should be tagged as one TIT-phrase, but at the same time, *Landgerichts Sulzbach* should also be tagged as a TIT-phrase. There are more similar cases in the corpus.

As the example in table 9 shows abbreviations which do not denote a title are problematic.

Interesting is the comparison of the tagging for the variations of *Nürnberg*. *Nürnberg* could correctly be tagged as LOC, likewise *Nürnberger* as PERS, but *Nürnberger (Gesandtschaft)* was labeled erroneously as PERS, too.

Kg	*O*	*O*	*B-TIT*
.	*O*	*O*	*I-TIT*

vs.

Kg	*O*	*B-PERS*	*B-TIT*
.	*O*	*I-PERS*	*I-TIT*
F	*O*	*I-PERS*	*O*
.	*O*	*I-PERS*	*O*

Table 7: Example 5

Gericht	*O*	*O*	*O*
Heroldsberg	*B-LOC*	*O*	*O*

vs.

Landrichter	*O*	*O*	*B-TIT*
und	*O*	*O*	*O*
Rechtsprecher	*O*	*O*	*O*
des	*O*	*O*	*O*
Landgerichts	*O*	*O*	*O*
Sulzbach	*O*	*O*	*O*

Table 8: Example 6

60	*O*	*O*	*O*
Pf	*O*	*O*	*B-TIT*
.	*O*	*O*	*I-TIT*

Table 9: Example 7

Bürgermeistern	*O*	*O*	*B-TIT*
und	*O*	*O*	*I-TIT*
Rat	*O*	*O*	*I-TIT*
der	*O*	*O*	*I-TIT*
Stadt	*O*	*O*	*I-TIT*
Windsheim	*B-LOC*	*O*	*I-TIT*

vs.

Bürgermeister	*O*	*O*	*B-TIT*
und	*O*	*O*	*I-TIT*
Räte	*O*	*O*	*I-TIT*
der	*O*	*O*	*I-TIT*
Städte	*O*	*O*	*I-TIT*
)	*O*	*O*	*I-TIT*
Nürnberg	*I-LOC*	*O*	*I-TIT*
und	*O*	*O*	*I-TIT*
Rothenburg	*O*	*O*	*I-TIT*

vs.

Bürgermeistern	*O*	*O*	*B-TIT*
und	*O*	*O*	*O*
Rat	*O*	*O*	*O*
der	*O*	*O*	*O*
Stadt	*O*	*O*	*O*
Windsheim	*O*	*O*	*O*

Table 10: Example 8

Quite inconsistent is the tagging of the phrase *Bürgermeistern und Rat/Räten der Stadt/Städte + LOC*. The example for this case shall not be further commented but ackownledged in table 10.

Additionally, the way the NE-tagger is currently trained many proper nouns are not recognized because they are not mentioned with a proper name but with a paraphrase or an indeterminate noun. Either the fact that these entities are not labeled can be accepted or the annotation guidelines should be changed, so that not only entities with proper names but also indeterminate nouns (without the titling of a personal name) are annotated. Since relations in the regesta texts, for instance between people and the location of town names, cannot be depicted, the author remarks that the format of the graph database as e.g. *Neo4j* is a good way of representing entities.

4.2 Historico-culturally

Even if the results of the tagger are by far not perfect, the next passage proves that the utilization of a NER can help with the finding and the analysis of entities in large corpus. Several patterns of names can be found in the corpus. In order to able to make some observations and statements about the (personal) names in the corpus, the chosen regesta texts first need to be arranged in time. All examined regesta texts originate from *Heft 14 Friedrich III., Nürnberg*. Its addition (1440-1149) reveals that the texts come from the late Middle Ages.[17]

Typical names in the Early Middle Ages were Germanic names. Because biblical names and names of hallows were in vogue in the later Middle Ages, only some Germanic-German names stayed, which were basically names of dynastic rulers (Debus, 2012). Such names could also be found and NE-tagged in the corpus at hand. These are names such as *Friedrich*, *Heinrich*, *Ludwig*, and *Konrad*. Names, in general, nowadays consist of an individual surname and an inherited family name. This system with two names developed in the Middle Ages in consequence of a need for further distinction (Debus, 2012). A person with a bipartite name in the corpus at hand is the *Bürger Georg Geuder (von Nürnberg)*. Other than the previous example, *Hans von Aich* is mentioned in a regesta without a family name but with a location supplement. This fact shows how one category of family names has developed: Some people - if they had no other significant characterization - were named after the town or city they came from (Debus, 2012). Interestingly, it catches ones eye that contrary to names of bourgeois, rulers and superior citizens are not mentioned with their family name. They could be clearly identified by their ruler title. In cases where a name of a ruler was not that clear, where it was possible that there existed more rulers with the same title and surname or where the local origin of the ruler was of importance, it was added: *Mgf. Albrecht von Brandenburg*, for instance. Rolker (Rolker, 2014) gives further explanations for rulers to not use their family names. As (Debus, 2012) tells, the medieval corporate society was relatively divided into aristocracy/knighthood, clergy, peas-

[17] The original document text comes from the late Middle Ages. The actual regesta text was written later, when the summaries of these documents were written for the RI. In any case, here, it is assumed that the names were not changed.

antry, and the slowly developing (urban) bourgeoisie. The NER mirrors this fact in the amount of tagged titles - *Kg.*, *Mgf.*, *Ritter*, *Bürgermeister*, *Gf.*, *Rat/Räte*, *Bürger* and much more. The pluralism of titles also shows the importance of them, especially when it came to jurisdiction.

4.3 Outlook

Since the performance of the NER-tagger was still not satisfying, several ways of improvement shall be presented as outlook in the following.

It can be supposed that the bad performance of the trained tagger can predominantly be attributed to an insufficient amount of training data. Hence, in order to improve the performance, the author suggests that the amount of tagged training data is enlarged. In this way, the model can learn more about the structure of the RI texts and recognize them in previously unseen data.

Basically, there are two types of NER - a probabilistic machine learning process, the type which was trained in the study at hand, and a lookup process. Probably, the combination of both would yield the best results. As Jurafsky and Martin (Jurafsky and Martin, 2017) outline in the list beneath, it is a common way to do a NER task by combining these two types. They go through the following steps and stages:

1. First, use high-precision rules to tag unambiguous entity mentions.
2. Then, search for substring matches of the previously detected names, using probabilistic string matching metrics [...].
3. Consult application-specific name lists to identify likely named entity mentions from the given domain.
4. Finally, apply probabilistic sequence labeling techniques that make use of the tags from previous stages as additional features.

Stanford offers the possibility of incorporating gazetteers into the final probabilistic NER process. The class *NERFeatureFactory* describes how this can be done[18]:

> The value can be one or more filenames (names separated by a comma, semicolon or space). If provided gazettes are loaded from these files. Each line should be an entity class name, followed by whitespace followed by an entity (which might be a phrase of several tokens with a single space between words). Giving this property turns on useGazettes, so you normally don't need to specify it (but can use it to turn off gazettes specified in a properties file).

Other approaches for improvement are the utilization of regular expressions, or - in the preprocessing of the corpus - to lemmatize and normalize (e.g. abbreviations as *Gf.* into *Graf*).

[18] https://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/ie/NERFeatureFactory.html.

References

Bartsch and Weber. 2015. Stanford named entity recognizer.

Darina Benikova, Chris Biemann, and Marc Reznicek. 2014. NoSta-D Named Entity Annotation for German: Guidelines and Dataset. page 1 of 8.

Friedhelm Debus. 2012. *Namenkunde und Namengeschichte : eine Einführung*, volume 51 of *Grundlagen der Germanistik*. Berlin.

Daniel Jurafsky and James H. Martin. 2017. *Speech and Language Processing. An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, third edition edition.

Christof Rolker. 2014. *Das Spiel der Namen. Familie, Verwandtschaft und Geschlecht im spätmittelalterlichen Konstanz*. PhD Thesis, Ostfildern.

Dieter Rübsamen and Manz Manz. 2013. Accumulated Register. Regesten Kaiser Friedrichs III. (1440-1493).