Ruprecht-Karls-Universität Heidelberg
Institut für Computerlinguistik

Bachelor's Thesis

Anonymization of German Medical Admission Notes

using Spelling Variant Detection and Named Entity Recognition

Name:	Phillip Dietrich Richter-Pechanski
Matrikelnummer:	3247875
Betreuer:	Stefan Riezler
Datum der Abgabe:	18. Januar 2018

Abstract

Medical texts are a vast research resource for medical and computational research. In contrast to newswire or wikipedia texts medical texts need to be de-identified before making them available for the public. I created a prototype for German medical text de-identification and named entity recognition using a three step approach. First I used well known rule-based models based on regular expressions and gazetteers, second I used a spelling variant detector based on Levenshtein distance, exploiting the fact that my medical texts contain semi-structured headers containing sensible personal data, and third I trained a named entity recognition model on out of domain data to add statistical capabilities to my prototype. Using a baseline based on regular expressions and gazetteers I could improve F_2-score from 78% to 85% for de-identification. My prototype is a first step for further research on German medical text de-identification and could show that using spelling variant detection and out of domain trained statistical models can improve de-identification performance significantly.

Zusammenfassung

Medizinische Texte sind eine große Forschungsressource für die medizinische und computerlinguistische Forschung. Im Gegensatz zu nachrichtlichen Texten oder Wikipediaartikeln müssen medizinische Texte vor ihrer Veröffentlichung de-identifiziert werden. Wir haben einen Prototyp für die De-Identifikation von medizinischen Texten und Named Entity Recognition erstellt. Zuerst benutzten wir bekannte regelbasierte Modelle, die auf regulären Ausdrücken und Gazetteers basieren, dann benutzten wir einen Spelling Variant Detector, der auf der Levenshtein-Distanz basiert und ausnutzt, dass unsere medizinischen Texte semi-strukturierte Briefköpfe besitzen, die sensible persönliche Daten enthalten. Drittens trainierten wir ein Named Entity Recognition-Modell auf out-of-domain Texten. Mit Hilfe einer Baseline, die auf regulären Ausdrücken und Gazetteers basiert, konnten wir den F_2-Score von 78% auf 85% für die De-Identifikation verbessern. Unser Prototyp ist ein erster Schritt zur weiteren Forschung auf dem Gebiet der De-Identifikation von deutschen medizinischen Texten. Zudem konnten wir zeigen, dass durch die Verwendung von Spelling Variant Detection und von statistischen Modellen die De-Identifikationsperformance deutscher medizinischer Texte signifikant verbessert werden konnte.

Contents

List of Figures

List of Tables

Introduction

Text based medical records are a vast resource for medical and computational research. Most of the records consist of sensible patient data linked to personal information such as names, addresses, contact numbers or insurance identification numbers. Supervised machine learning approaches based on statistical and neural models in natural language processing demand large amounts of data. Before medical records can be processed for information extraction by non-medical staff they need to be de-identified. Manual de-identification can be costly, time consuming (Douglass et al. 2004) and unreliable (Neamatullah et al. 2008), furthermore there are just a few people authorized to view the raw data. This raises the need for an automatic de-identification processes for medical records.

While there are non-German de-identified data sets for training and evaluation freely accessible like the *i2b2 2014* de-identification challenge dataset and the *MIMIC* de-identification dataset[1], shared data sets for German medical records are still lacking. Due to this the aim of this project is to build a named entity recognition (NER) and de-identification tool for German medical records.

For the accomplishment of this task I had the opportunity to use a raw corpus of around 180 000 medical admission notes in binary MS DOC format with a total of around 132 million tokens due to the cooperation of the Section of Bioinformatics and Systems Cardiology at the Klaus Tschira Institute for Integrative Computational Cardiology under the supervision of Christoph Dieterich with the Department of Computational Linguistics at Heidelberg University supervised by Stefan Riezler and the Database Systems Research Group at the Heidelberg University supervised by Michael Gertz.

Due to the total lack of annotated German medical records I developed a three step approach based on rule-based methods and an out of domain trained statistical model. First I combined a rule-based NER model and a rule-based spelling variant detection algorithm based on Levenshtein distance, second I added a statistical NER model trained on German non-medical texts. With this approach I significantly improved F_2-score in comparison to de-identification algorithms solely based on

1. i2b2: `https://www.i2b2.org/NLP/DataSets/Main.php`
MIMIC: `https://mimic.physionet.org/`

rules.

This thesis is structured as follows: in the first chapter I specify the meaning of the terms anonymization, pseudonymization and de-identification; furthermore I specify which tokens need to be removed from a medical admission note to comply with legal standards; next I will present related work on NER and de-identification of medical texts in general and German medical texts in particular. In addition I list some related work in the field of out of domain NER models. The third chapter presents a quantity and a quality analysis of my data set and defines a test set for evaluation; next I describe the theoretical background of my approach. I give a short introduction to statistical NER, spelling variant detection and rule-based NER; thereupon I present the implementation of my algorithm for NER and de-identification on my data set. In addition I present my development environment and the preprocessing steps; eventually I describe my experiments and the evaluation task. After explaining my baseline I present my scores and statistical tests. I then present my evaluation results for the experiments. Finally I evaluate the significance of my final NER and de-identification model. The last chapter gives a résumé of the work accomplished and shows future work possibilities in this topic.

1 Anonymization, Pseudomization and De-Identification

To analyze medical data the consent of the patient is the legal basis generally preferred by law. Due to the vast amount of data, this is typically not possible to fulfill (Schlunder 2015). If there is no individual consent medical texts can only be analyzed without legal restrictions if the personal data is fully removed (Schlunder 2015). The task of removing personal data is described with several different terminologies. Though often used in a similar context de-identification, pseudonymization and anonymization are distinct techniques.[2]

In March 2018 the General Data Protection Regulation (GDPR) will replace

2. https://iapp.org/news/a/looking-to-comply-with-gdpr-heres-a-primer-on-anonymization-and-pseudonymization/

the twenty year old Data Protection Directive (DPD) in the European Union. Unfortunately the regulations/directives neither give a consistent definition about the three terms, nor do they specify what tokens need to be removed to comply with the DPD/GDPR.[3] Hence I will use definitions from the GDPR, DPD, International Organization for Standardization (ISO) and the U.S. Health Insurance Portability and Accountability Act (HIPAA)[4] to ensure a clear objective for my task.

De-identification is the most general term. **De-identification** describes the process of removing the association between a set of identifying data and the data subject.[5] According to the GDPR data processed via **anonymization** are

> *data rendered in such a way that the data subject is not or no longer identifiable.*[6]

Anonymized data can **never** be re-identified.[7] In contrast **pseudonymization** defines

> *the processing of personal data in such a way that the data can no longer be attributed to a specific data subject without the use of additional information.*[8]

In this case the key data need to be kept separate from the pseudonymized data, to be used more liberally. As I did not make a clear distinction between pseudonymization and anonymization, I used the most general term de-identification in the rest of this project.

The descriptions of the terms de-identification, anonymization and pseudonymization are very shallow. Both the European DPD and GDPR as well as the different legal rules in the 16 states of Germany lack the specification which tokens need to be removed from medical texts, to comply with the laws.[9] This is why research studies

3. Steve Touw, CTO and Co-founder of the data-science company Immuta called the guidlines in the GDPR *very blurred.*

4. `https://en.wikipedia.org/wiki/Health_Insurance_Portability_and_Accountability_Act`

5. `https://en.wikipedia.org/wiki/De-identification`, (ISO/TS 25237:2008)

6. `http://www.eugdpr.org/`

7. EU Directive 95/46/EC

8. GDPR, Article 4

9. Landesdatenschutzgesetz, Abschnitt 1, Paragraph 3, 6 and Art. 9 – EU-DSGVO – Verarbeitung besonderer Kategorien personenbezogener Daten

of non-German medical texts, e.g. (Douglass et al. 2004; Neamatullah et al. 2008; Uzuner et al. 2008; Scheurwegs et al. 2013), mostly stick to the definition in the HIPPA which in contrast identifies 18 personal health information types (PHI).

1.1 What tokens to de-identify?

With the definition of the PHI tokens in the HIPAA I could limit my de-identification approach to a fixed set of tokens. Most of the existing de-identification systems target only several types of identifying information listed as PHI (Meystre et al. 2010). In my admission notes the following PHI needed to be recognized and de-identified:

- Names
- Geographic data
- All elements of dates
- Telephone numbers
- FAX numbers
- E-mail addresses
- Web URLs

In addition my algorithm de-identified salutations (Herr, Frau, ...) and academic titles (Dr., Prof., ...). Due to simplicity reasons my algorithm de-identified by default not only tokens related to patients but to all individuals, like physicians or assistants.

2 Related Work

Machine driven de-identification of medical texts had been a research task since the 1990s. At first all approaches relied basically on rule-based algorithms. These systems used manually defined patterns as regular expressions and gazetteers to identify PHI (A Beckwith et al. 2006; Berman 2003; Gupta, Saul, and Gilbertson

2004; Neamatullah et al. 2008; Ruch et al. 2000; Thomas et al. 2002; Sweeney 1996; Taira, Bui, and Kangarloo 2002; Yuwono, Ng, and Ngiam 2016). (Sweeney 1996) was the one who laid the foundation for automatic de-identification tasks on medical texts. He processed a data set of around 3000 medical records derived of around 275 patients. Most of the data had been machine samples with semi-structured data. Most texts contain a header, a salutation and well-formed sentences. Sweeney used gazetteers and templates for PHI recognition.

In 2002 there had been research using semantic restrictions and maximum entropy classifiers to identify PHI sequences (Taira, Bui, and Kangarloo 2002). In the same year (Thomas et al. 2002) implemented an augmented search and a replacement algorithm for medical text de-identification. (Neamatullah et al. 2008) used medical and hospital specific information for automated pattern-matching and de-identification. (Yuwono, Ng, and Ngiam 2016) used spelling variant detection for de-identification of unstructured medical texts, based on structured PHI entities in a database. This approach was part of my strategy as well, as my data set contained a semi-structured header containing important PHI entities.

(Wellner et al. 2007) finally combined traditional rule-based approaches and state of the art supervised machine learning tools to recognize PHI tokens in medical texts. They tested their system using NER tools based on conditional random fields (CRF) and hidden Markov models. During evaluation of a large data set the CRF outperformed the hidden Markov model in the de-identification task. In addition the research revealed that already small training sets with PHI sequences can improve the performance of the de-identification tool significantly (Wellner et al. 2007; Scheurwegs et al. 2013).

After (Sweeney 2002) defined a measure for evaluating the quality of de-identification (Gardner and Xiong 2008) used this measure for his de-identification tool kit using mainly CRF models. They addressed as well the problem of defining PHI and de-identification as a legal aspect. They stated that full compliance with HIPAA is an almost unrealizable task as next to the 18 defined PHI types the HIPAA states that "any other unique identifying number, characteristic or code" needs to be de-identified. That is why they focused on partial de-identification, ignoring not directly linked personal data like age and gender specifications.

A Support Vector Machine (SVM) classifier and context aids had been the

approach of (Uzuner et al. 2008) for their de-identification algorithm. As a baseline they used rules and gazetteers. Their system perfomed well even on fragmented medical texts. (Scheurwegs et al. 2013) used an SVM for PHI recogmition, as well. They used a small training set of 100 Dutch medical texts for their task.

In 2013 researchers discovered that rule-based methods reach better recall, while machine-based methods reach better precision rates. They combined both approaches into a de-identification framework. After pre-processing of the medical texts they performed a sensitivity extraction process using rules from patterns (including Levenshtein distance), dictionaries and CRF models from the Stanford Core NLP library trained on medical training sets. This process focused on recall performance. In the next step they performed a false positive filtering step, improving the precision of the framework. Here they used a SVM classifier trained on the texts created after the sensitivity extraction step. The SVM classifier just focused on the PHI with low precision performance (PLZ, DATE, PHONE). (Ferrandez et al. 2013) finally chose F_2-score, to weight recall in evaluation higher.

A lot of these approaches operating on non-German medical texts combined rule and dictionary based strategies with statistical models, typically NER. For further information about de-identification systems Stubbs extensively reviewed existing de-identification systems (Stubbs, Kotfila, and Uzuner 2015).

More recently there were approaches based on artificial neural networks. One approach did not need any manually curated features to avoid extensive feature engineering (Dernoncourt et al. 2016). (Lee et al. 2016) presented a neural network approach which combined a network with human augmented features. With a combination of CRF classifiers and recurrent neural networks (Zengjian Liu et al. 2015) developed a de-identification system. Martinez lately developed a system for NER by training word embeddings on Spanish medical texts (Soriano and Pena 2017).

Due to the total lack of shared German data the use of supervised learning NLP tools is restricted as they need training data to work properly (Starlinger et al. 2017). There are several local researches on German medical texts, but no publications are available. There are some commercial tools, provided by Averbis

and Statice.[10] Out of a larger multilingual data set (Ruch et al. 2000) used a few German examples for de-identification with a semantic lexicon. Active learning had been the main strategy of Joachim Wermter and Katrin Tomanek in 2006.[11]

In this project I applied an out of domain trained NER model for my German medical admission notes. (Ciaramita and Altun 2005) investigated the performance of a supervised NER model on data from a different domain than the training data. They discovered a decrease in F_1-score using out of domain training data. They did not make a distinction between recall and precision though. (Persson 2017) investigated the influence of excluding out of domain data during NER training. His results showed no significant increase in F_1-score after exclusion. There had been several approaches in the area of domain adaption to improve out of domain training performance (Guo et al. 2009; Kulkarni, Mehdad, and Chevalier 2016; Sun, Grishman, and Wang 2016). As I used existing NER systems in my project and used the results rather as a proof of concept I did not focus on the topic of model design.

3 Data

The medical data set used in this project consists of medical admission notes as binary MS-DOC files containing texts from the domain of cardiology. Medical admission notes (Arztbriefe) vary a lot in scope and structure between different medical areas. They are supposed to be short and concise. Next to personal data like the patients name, address and birth date, the notes shall contain past and current diagnoses. In addition the medical patient history and planned medical examinations are described. If accomplished, results of laboratory examinations are as well part of an admission note.[12]

Most of the admission notes in our data set contain an introductory sentence stating that the note is still incomplete. This is true especially for the ICD-10

10. `https://averbis.com/`, `https://www.statice.io/`
11. `http://www.gmds2006.imise.uni-leipzig.de/Vortraege/128.pdf`
12. For more information see: `https://de.wikipedia.org/wiki/Arztbrief#cite_note-1`

terms[13], which are not part of this data set.[14] In addition, the structure and length varies a lot. Some letters consist of half a page some consist of three to four pages.

Still all medical notes share a basic structure. The majority of the admission notes in this project have the following basic structure:

- Header
  - Adressee
  - Sender
  - Patients name and address
- Salutation
- Summary

In addition the notes contain a subset of the following subsections:

- Diagnosis
- Cardiovascular risk factors
- Allergies
- Anamnese
- Physical examination (Körperlicher Untersuchungsbefund)
- Laboratory data (some in tabular structure)
- ECG
- Recommended therapy

13. International Statistical Classification of Diseases and Related Health Problems. For more informations see: `https://www.dimdi.de/static/de/klassi/icd-10-gm/index.htm`.

14. If they were part of the data, we could use the data as a training set for ICD-10 recognition tasks.

Year	Notes	Non-blank lines	Tokens
2004	2 272	302 997	1 823 801
2005	2 432	338 047	2 020 605
2006	2 185	300 386	1 872 093
2007	2 163	371 703	2 381 309
2008	1 989	364 386	2 427 136
2009	11 029	1 368 488	8 462 008
2010	44 099	5 095 453	30 330 830
2011	44 969	5 327 624	31 887 364
2013	22 426	2 515 247	14 414 940
2014	45 087	5 845 171	33 803 759
2016	2 249	516 175	3 335 795
Total	180 900	22 345 677	132 759 640

Table 1: Quantity analysis of medical admission notes by year

The amount of text in each subsection is varying. The subsections contain free unstructured text, sometimes tables or images. Occasionally subsections are titled differently, but contain similar information, e.g. therapy/medication. Often terms are abbreviated, e.g. CRF/Cardiovascular risk factors.

The notes are concluded by a salutation and the names of the physicians involved. In figure 1 you can find a dummy extract showing a part of a typical admission note structure. Additionally table 1 shows a quantity analysis of the data set.

3.1 Test Set

The test set used in this project consists of 16 randomly chosen samples from my medical admission notes corpus. They had been labeled manually by authorized persons using named entity (NE) tags defined in table 8. The class distribution is listed in table 2. For the annotation task I set up a WebAnno[15] environment with a pre-defined project containing all of my named entity classes and an annotation guideline describing my entity classes (Yimam et al. 2013). For evaluation I used the two column CoNLL 2002 format.

The notes in the test set contain a total of 14 134 tokens from which 680 are named

15. `https://webanno.github.io/webanno/`

Sehr geehrte Kollegen,
wir berichten über Ihren Patienten Herrn Mustermann, geboren am 01.12.1956, wh. Musterstraße, 12345 Musterstadt der sich am 12.12.2010 in unserer Ambulanz vorstellte.
Diagnosen:
Aktuell: Operabilität mit leicht erhöhtem Risiko gegeben Z.n. Herzkatheter-Untersuchung, zuletzt 01.04.2009 (anamnestisch) Z.n. weiterer Eingriff 12/09 (in Beispielhausen), Versuch der ... Anamnestisch OP bei ... 10.01.2010
Anamnese:
Die Vorstellung des Patienten erfolgte zur präoperativen kardiologischen Abklärung vor geplanter Operation, ...

Figure 1: Extract of a raw medical admission note

entities. For a representative and consistent evaluation of multiclass recognition and binary PHI recognition I excluded the header from my test set as it was fully parsed for spelling variant detection. This would always lead to a F_2-score = 100% for binary PHI recognition in the header. Due to the exclusion there were no URI (e.g. web addresses) and EMAIL tags in the test corpus as they just appeared in the header.

4 Theoretical Background

As the task of de-identification of German medical texts is still little researched, my project is based on de-identification approaches of non-German medical texts. Insufficient amount of training data available motivated me to use rule-based methods for my task in the first place. Still to gain experience in the field of machine learning approaches for German medical texts I exercised a proof of concept by applying an out of domain trained NER model on my data (Ciaramita and Altun 2005).

The medical admission notes are semi-structured. This makes it possible to extract PHI tokens from the header of a note and to remove all appearances of this

Named Entity	Tokens
DATE	241
PER	165
LOC	104
TITLE	75
SALUTE	52
PHONE	26
PLZ	15
ORG	2
URI	-
EMAIL	-
Total	680

Table 2: Class distribution in test set. For further explanation see table 8

token in the following unstructured text. Because PHI tokens in the header can appear in spelling variants in the text, I used a spelling variant detector based on the Levenshtein distance to match PHI tokens (e.g Müller vs. Mueller) (Yuwono, Ng, and Ngiam 2016).

In addition to the latter approach I used rules and gazetteers containing patterns and regular expressions. For a smooth integration in my project, I used Stanford RegexNER for this task (Manning et al. 2014; Sweeney 1996).

My machine learning approach was based on two assumptions; all PHI tokens were named entities and some named entities had a similar pattern in cross domain data sets. Because of the latter assumption I trained a NER model on an out of domain data set.

4.1 Statistical NER on Medical Texts

Previous research showed that out of domain trained NER models had a similar or worse F_1-score than in domain trained models (Ciaramita and Altun 2005; Persson 2017). As these publications used newswire data sets or just very small scientific data sets as evaluation sets they might not be representative for my tasks of NER and de-identification on medical data sets. In addition past research did not make a distinction between the performance of specific named entities and scores like precision and recall.

Parameter	Value
useClassFeature	true
useWord	true
useNGrams	true
noMidNGrams	true
useDisjunctive	true
maxNGramLeng	6
usePrev	true
useNext	true
useSequences	true
usePrevSequences	true
maxLeft	1
useTypeSeqs	true
useTypeSeqs2	true
useTypeySequences	true
wordShape	chris2useLC
printFeatures	true

Table 3: Parameters for Stanford NER training. For more information about training and parameters see [17]

After investigating the performance of NER tools on out of domain data for German in my former task "Evaluation of German NER Tools"[16] I examined the performance of these tools on German medical texts. I identified the German Stanford NER as the best performing tool trained on out of domain data and selected this tool for my task for NER on German medical texts. Stanford NER is based on a CRF model. A CRF is a conditional sequence model representing the probability of a sequence of hidden states given an observations (Finkel, Grenager, and Manning 2005). In a previous task (Wellner et al. 2007) identified CRF systems as best performing method for medical text de-identification.

The most convincing Stanford NER model was trained on a concatenation of three corpora open to the public and typically used in German NER research. The parameters used for training are listed in table 3.

German **CoNLL 2003**: Selected texts from German newspaper Frankfurter

16. https://github.com/MaviccPRP/ger_ner_evals/

LOC	ORG	PER
4 363	2 427	2 773

Table 4: Class distribution in CoNLL 2003

LOC	ORG	PER
12 791	9 889	12 423

Table 5: Class distribution in GermaEval 2014

Rundschau. The training data consists of 206 931 tokens in 12 705 sentences. The named entity (NE) classes are distributed as in table 4. The **GermEval 2014** data set contains text from German Wikipedia articles and online news texts. The training data consists of 24 000 sentences. The data set contains over 590 000 tokens. The NE classes are distributed as in table 5. In addition I used texts from the **European Parliament** annotated by Sebastian Pado following the CoNLL 2003 guidelines[18]. The class distribution is listed in table 6.

Still there were some major challenges in research on NER on medical texts. Our medical admission notes had a different structure and terminology than newswire, Wikipedia and political texts. Most of the time medical texts are free unstructured texts, which are sometimes semi-structured. The texts contain plenty of non standardized and ambiguous abbreviations and a varying and sometimes even locally specific terminology. Furthermore these texts often include lists and tables consisting of medical terminology and numerical laboratory results. In contrast to newswire, Wikipedia and political texts medical texts are discontinuous containing rather short sequences of sentences interrupted by structured and semi-structured sections.

In my previous task I could not automatically evaluate the NER model on a

18. `https://www.nlpado.de/~sebastian/software/ner_german.shtml`

LOC	ORG	PER
451	476	385

Table 6: Class distribution in European Parliament

LOC	ORG	PER
18	25	22

Table 7: Class distribution of one letter test set

larger medical text test set, because of the lack of annotated data. Due to this limitation and the lack of time, I chose randomly one admission note and manually annotated the file with three NE classes (PER, LOC, ORG). Thus the resulting scores are rather hints and require further work.

The test data contains 1 049 tokens with a class distribution shown in table 7. For the sake of simplicity I evaluated per token, not per entity sequence. To get the harmonic mean of both precision and recall, I evaluated my model using the F_1-score. My model reached its best F_1-score for the PER class with 84%. The LOC class had been identified with a F1-score of 54%. The ORG class reached a considerable lower score of 11%.[19]

The results of the PER class supported my assumption that some named entities have a similar pattern in cross-domain data sets. Tokens of first names and surnames in newspaper texts, Wikipedia articles or medical admission notes have a similar structure. To a lesser extend this is true for the LOC class. The class ORG performs worse, possibly because the GermaEval and CoNLL corpora contained rather political and cultural organizations than medical ones. (Vereinte Nationen vs. Uniklinik Heidelberg) I will further investigate this class later in this project.

4.2 Rule-based NER

My rule-based method used Stanford RegexNER for de-identification because of the well documented API (application programming interface) and a smooth integration into the Stanford Core NLP pipeline (Manning et al. 2014). RegexNER is a pattern-based tool for NER. It takes plain text files as gazetteers. Each file contains two columns where the first column contains text to match and the second column contains labels to assign. The tokens can contain regular expressions. Regular expressions are a sequence of characters defining a search pattern.

19. For more infos see: `https://github.com/MaviccPRP/ger_ner_evals/blob/master/reports/report.pdf`

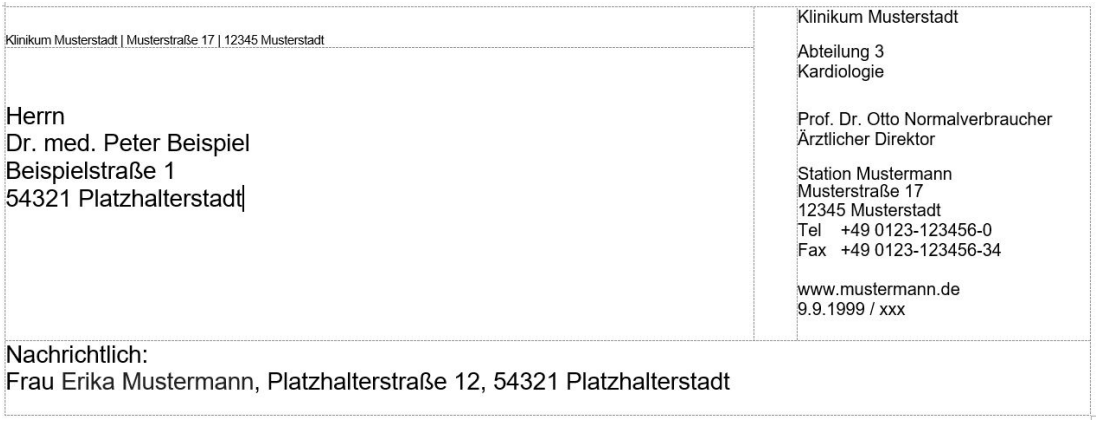

Figure 2: Example header in a medical admission note

Rule-based methods require little or even no training data. In addition they can be quickly altered to improve performance. Rule-based methods play an important role in de-identification of medical texts, often combined with statistical/neural NER models. Relatively small changes in regular expressions or gazetteers can have a great impact on the system performance (Meystre et al. 2010). Context knowledge and the exploitation of regional or local knowledge can vitally improve the system performance. Rule-based NER is especially well established on narrow domains and some specific named entity like postal codes and emails addresses, as these texts have a relatively fixed pattern. The main disadvantage of rule-based methods is their dependence on manually created lists, at the cost of time consuming work by human experts. In addition these rules do not guarantee generalizability (Meystre et al. 2010). Small irregularities in token patterns can make very sophisticated rules useless.

4.3 Spelling Variant Detection

The second rule-based de-identification step exploited the fact that a semi-structured header was part of most of my medical admission notes. This header mostly contains the name and address of the patient, the name and address of the recipients of the note and the contact information of the clinic. Image 2 shows an example header. Often the most important cell in this header is marked with the key word *Nachrichtlich*. As this key word flags the patients contact information, the cell

contains very sensible personal data. These tokens and their variants need to be de-identified in a de-identification task. As the key word *Nachrichtlich* is not included in all headers by default my approach de-identified all tokens which appear in the header, including clinical and physician information, except a pre-defined list of stopwords. I adapted the approach of (Yuwono, Ng, and Ngiam 2016) who used PHI tokens from a structured part of a medical record to find appearances of these tokens in the rest of the medical text. They used minimum edit distance to identify spelling variants in the texts (Yuwono, Ng, and Ngiam 2016; Wagner and Fischer 1974). The minimum edit distance ratio R is computed as follows:

$$R = \frac{d}{min(|n|, |w|)} \tag{1}$$

Where d is the minimum edit distance of the tokens n and w. As long tokens have a higher probability of being misspelled than short ones, R is used to take into account the length of a string. If R is under a threshold of 0.333 w will be de-identified.

(Yuwono, Ng, and Ngiam 2016) replaces each PHI token by its PHI type defined in the structured part of the medical text (e.g. PNAME, PID). I pre-annotated the admission notes via a rule-based and statistical named entity recognizer and thus was able to use the named entity class of each detected PHI token as a placeholder.

5 Implementation

Figure 3 shows my implementation of the de-identification tool. After preprocessing, parsing the header and annotating the admission note with a rule-based RegexNER and a statistical NER model the algorithm starts the de-identification step.

1. It will replaces all spelling variants of tokens found in the header with its NE label or 'O' if it is not recognized as a NE by the NER models.

2. If the token is not a spelling variant but annotated as a NE it will be replaced by its NE label, too.

3. If the token is neither a spelling variant nor labeled with a NE class, the algorithm checks if the token contains location suffixes, pre-defined in a list.

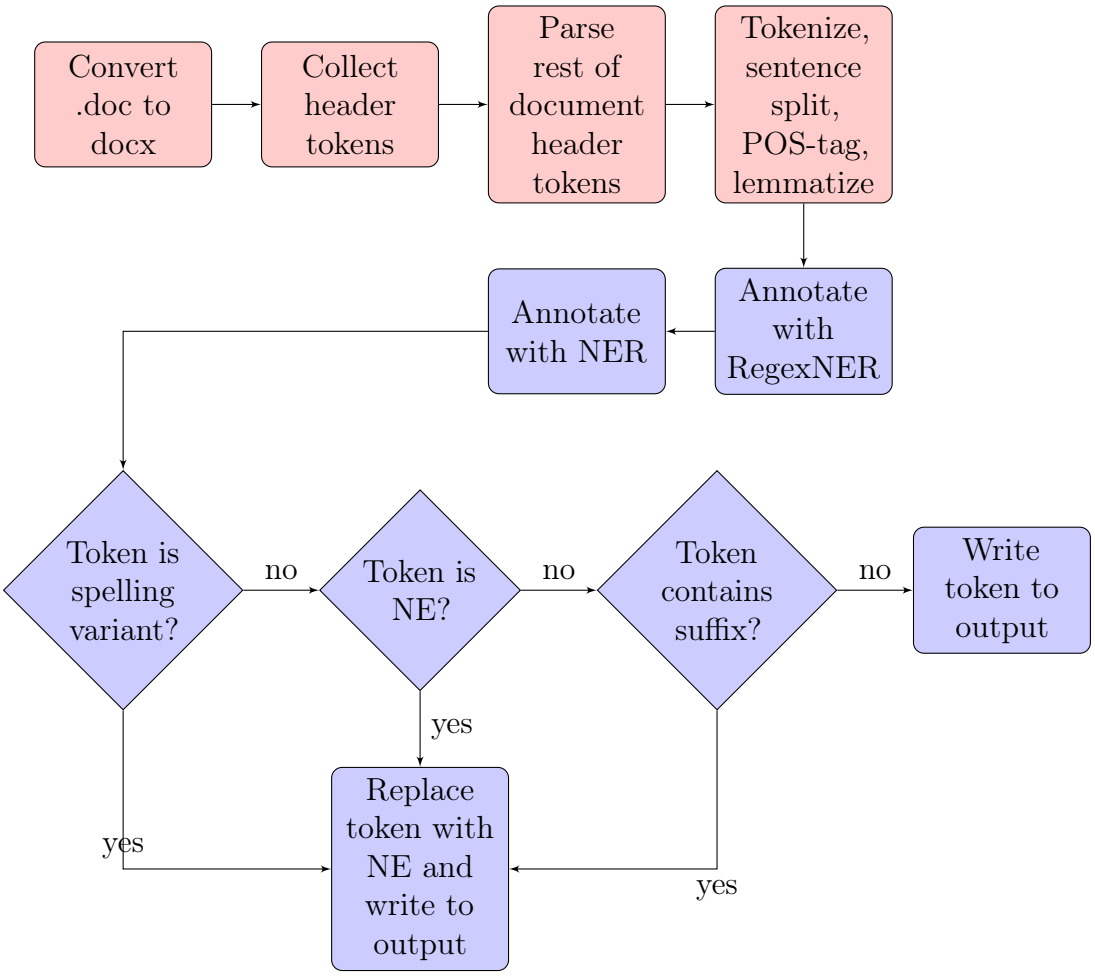

Figure 3: Algorithm of the de-identification tool

Sehr geehrte Kollegen,

wir berichten über Ihren Patienten <SALUTE> <PER>, geboren am <DATE>, wh. <LOC>, <PLZ> <LOC> der sich am <DATE> in unserer Ambulanz vorstellte.
Diagnosen:
Aktuell: Operabilität mit leicht erhöhtem Risiko gegeben Z.n. Herzkatheter-Untersuchung, zuletzt <DATE> (anamnestisch) Z.n. weiterer Eingriff <DATE> (<LOC>), Versuch der ... Anamnestisch OP bei ... <DATE>
Anamnese:
Die Vorstellung des Patienten erfolgte zur präoperativen kardiologischen Abklärung vor geplanter Operation, ...

Figure 4: Extract of a de-identified nedical admission note

- If the token contains a suffix it will eventually be de-identified and replaced by <LOC>.

4. If the token is neither a NE, a spelling variant nor contains a suffix it will be directly written to the output.

After execution the algorithm resulted in an output as shown in figure 4.

5.1 Tools

My algorithm is based on Java 8 and Python 3 as this guarantees easy UTF-8 string handling and a smooth integration of my used libraries. To convert the binary MS DOC files to MS DOCX I used the LibreOffice `convert-to` command line tool. With Apache POI XWPF I extracted structured information from the header tables. I used the Stanford Core NLP pipeline to integrate my annotation, de-identification and preprocessing steps like tokenization, sentence splitting, lemmatization and part-of-speech tagging (Manning et al. 2014; Ferrandez et al. 2013). After preprocessing I added my rule-based NER model to the pipeline using Stanford RegexNER (Manning et al. 2014). Next I integrated my out of domain trained Stanford NER model (Faruqui and Padó 2010). For experiments and evaluation I

used the Python library scikit-learn. For my statistical tests I used mlxtend and Sebastian Pado's SIGF V2 (Padó 2006).

5.2 Preprocessing

In the first preprocessing step I converted the binary MS DOC files to MS DOCX files. This step ensured the possibility to parse structured tables with Apache POI XWPF. This was especially important for my spelling variant detector as I needed to parse all tokens in the header table. In a next step I preprocessed the raw strings for further processing. My tokenizer was based on a pre-trained German model provided by the Stanford NLP Group. I used default parameters for tokenization. Due to a considerable amount of mistakes of the sentence splitter on my medical admission notes, I used it with the *isOneSentence* parameter. This kept the whole admission note as one string. This configuration prevented a reasonable amount of regular expressions from my RegexNER model to fail due to wrong sentence splitting. E.g. some sentences would be splitted after street abbreviations (Hauptstr.), thus the house number would be part of the next sentence. The pre-trained German pos-tagger and lemmatizer models were used with default parameters. All these steps were obligatory for the Stanford NER and RegexNER models.

6 Experiments and Results

6.1 Baseline

For evaluation purposes I used a rule-based approach as a baseline (Uzuner et al. 2008). It used Stanford RegexNER for multiclass NER and binary PHI recognition. The effect of spelling variant detection was evaluated in binary PHI recognition as the detector was not labeling additional named entities and only decided weather to de-identificate a PHI tag or to keep it. I built my RegexNER model with gazetteers containing plain text for first names, surnames and German cities and towns furthermore with gazetteers containing German street names with regular expressions for street names, phone numbers, dates etc.

- German towns and villages (11 739 tokens)
- German and international surnames (31 983 tokens)
- German and international first names (20 562 tokens)
- German street names including regular expressions to recognize house numbers and abbreviations (96 100 tokens)
  `Haup[sS]tr[.]?(a(ß|ss)e)?  [0-9]{0,3}[a-zA-Z]?[.,]?`
- International first names (20 561 tokens)
- Regular expressions for postal codes, URIs, email addresses, dates, phone/fax numbers, salutes and titles

To recognize unknown towns and streets I used additionally suffix gazetteers:

- street suffixes: straße, str., strasse, Straße, weg, Weg, rhain, Rhain, gasse, Gasse, allee, Allee, etc.
- town suffixes: heim, berg, ingen etc.

If a token contained one of these suffixes, it was replaced by a LOC placeholder. My baseline recognized a set of NE classes shown in table 8.

6.2 Scores and Statistical Tests

For the sake of simplicity I evaluated my binary PHI and multiclass NER de-identification tool per token. This means that if my de-identification tool recognized one token in a two token sequence, I counted this token already as a true positive while the token not recognized was counted as a false negative. Figure 9 shows an example multiclass NER classification.

The widely used overall accuracy measure treats all classes in the same way. The prediction of smaller classes is mostly shadowed by the prediction of much bigger classes. Moreover accuracy does not take into account that my data set was highly imbalanced. There were a lot of non-PHI tokens in my data set but just a few but very important PHI tokens. Not finding a PHI had a much higher cost than de-identifying a non-PHI in my task, because sensible personal data could be visible

Class	Description	Example
PER	Surnames, first names, other names of individuals	(Maria, Schulz ...)
LOC	Cities, countries, other geographic locations	(Haupstraße, Berlin)
SALUTE	Part of the salutation before a name. Usually refers to a gender	(Frau, Herr, Herrn)
EMAIL	all proper mail addresses based on the RFC standards	(max@mustermann.de)
PHONE	All kinds of fax/phone numbers	(012 - 34 5677)
URI	All proper uri addresses based on the RFC standards	(www.example.xyz)
DATE	all kinds of date representations	(01.12.2015, Jan. 1999)
PLZ	All proper postal codes based on the standards in Germany	14123
TITLE	All kinds of academical titles	(Dr., Dr. med.)

Table 8: NE classes in medical admission notes incl. description and examples

Ich	heiße	Johannes	Paul	Mustermann	.
O	O	PER	PER	O	O

Table 9: Example classification of multiclass NER. There is a three token sequence named entity (Johannes Paul Mustermann) in the sentence. My classifier recognizes just Johannes Paul. Thus I got two true positives, one false negative and three true negatives.

in already processed text. Further there are widely used scores like precision, recall and its harmonic average F-score. Recall is the fraction of relevant samples that have been recognized over all relevant samples. Precision is the fraction of relevant samples among the recognized samples. Having a high recall typically decreases the precision (K. Buckland and Gey 1994). Since I wanted to find as much as possible PHI tags in my corpus improving recall was my main goal thus keeping precision as high as possible not to loose any relevant information. To take into account the high relevance of recall, in addition to recall and precision I used F_2-score for calculating a single evaluation score for my classifiers (Ferrandez et al. 2013). In contrast to F_1-score F_2-score weights recall four times higher than precision.

$$F_2 = \frac{5 * precision * recall}{(4 * precision + recall)} \tag{2}$$

To rate the quality of my classifiers I used Cohen's kappa coefficient. This coefficient is useful for highly imbalanced data sets. The kappa coefficient compares the observed accuracy (p_o) with the expected accuracy (p_e) (rbx 2014). Expected accuracy takes class distribution into account and represents the score that any random classifier would achieve on a given confusion matrix. It is directly related to the amount of instances of each class.

$$p_e = \frac{\sum_{i=1}^{k} n_{i.} * n_{.i}}{N^2} \tag{3}$$

$n_{.i}$ represents the total sums of the predicted class in column i. $n_{i.}$ represents the total sums of the true classes in row i. N represents the total amount of all samples. Observed accuracy represents the fraction of all true classified samples by all entries in the confusion matrix.

$$p_o = \frac{\sum_{i=1}^{k} n_{ii}}{N} \tag{4}$$

n_{ii} represent the true classified samples in a confusion matrix. Kappa is calculated using both accuracy scores:

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{5}$$

κ-score	amount of agreement
$0 - 0.2$	slight
$0.21 - 0.4$	fair
$0.41 - 0.6$	moderate
$0.61 - 0.8$	substantial
$0.81 - 1$	almost perfect

Table 10: κ-statistics by Landis and Koch

Cohen's kappa is always less than or equal to 1. Classifiers with values of 0 or less are useless. There is no standardized way to interpret the kappa values. Still Landis and Koch provided a scale to interpret the performance of a classifier, see table 10 (Landis and Koch 1977).

To determine if my full featured classifier significantly differed from my baseline classifier I used McNemars χ^2-test for multiclass and binary PHI recognition and approximate randomization for binary PHI recognition (McNemar 1947; Padó 2006).

McNemars test is based on a dichotome characteristic like a $2x2$ truth table.[20] The test is applied to compare the output of two classifiers on paired data as I measured performance of two classifiers on the same test set. The test counts the unique errors made by each classifier. There are four possible outcomes of two classifiers:

- Both classifier predict correctly.
- Both classifier misclassify.
- Classifier A predicts correct, Classifier B misclassifies.
- Classifier B predicts correct, Classifier A misclassifies.

McNemar focuses on the last two values. The four outcomes can be represented in a truth table like in table 11. In this contigency table the test would be applied on the entries b and c. Whereas b represents a sample which had been labeled right

20. `https://de.wikipedia.org/wiki/McNemar-Test`

Classifier B	Classifier A: +	Classifier A: −
+	a	b
−	c	d

Table 11: $2x2$ truth table for McNemar's testing

by Classifier B and wrong by Classifier A, vice versa for c. The test is computed as follows:

$$\chi^2 = \frac{(|b-c|-1)^2}{b+c} \tag{6}$$

McNemar calculates the fraction of the differences of the classifier where they predict different and the sum of these values. If $\chi^2 = 0$ both classifier perform similar. If $\chi^2 > 0$ performance is different. The result can be compared to the values in a χ^2 table with one degree of freedom and a pre-defined p-value. McNemar's test says nothing about which classifier performs better or worse. However this can be deduced from the error rates in truth table 11.

Approximate randomization is widely used in NLP and de-identification tasks (Yeh 2000; Noreen 1988; Dernoncourt et al. 2016). In contrast to exact randomization it does not iterate over the whole data set but just samples from it (Morgan, no date). Under the null hypothesis two classifiers produce the same results. Randomly shuffling strata statistics of two classifier with equal probability and recompute the aggregate statistic creates an approximate distribution of the test statistics under the null hypothesis (Riezler and Maxwell 2005). For approximate randomization I stratified the results of my baseline and full featured classifier. A p-value is computed by dividing the number of trials where my shuffled strata statistic (in my case F_2) is larger or equal to the actual statistic by the number of iterations.

$$p_{value} = \frac{r+1}{R+1} \tag{7}$$

Lowercase r represents the amount of times the strata statistic is greater than the aggregated statistic. Capital R represents the number of iterations, which typically is 1 000 (Morgan, no date). The test makes no assumptions on the distribution but it assumes that there is independence between the shuffled samples.

Named Entity	Precision	Recall
PER	0.63	0.97
LOC	0.51	0.27
ORG	0.01	1.00

Table 12: Preliminary NER on medical test set

Model	ANON	KEEP
Baseline	0.78	0.98
Baseline + Spelling Variant	0.83	0.98
Full Featured	0.85	0.98

Table 13: PHI Recognition F_2-score

6.3 Evaluation

6.3.1 Preliminary Work

In my initial task "Evaluation of German NER Tools"[21] I had a very small test set to evaluate the out of domain trained named entity model. Thus I evaluated the model on my extended test set containing 16 admission notes, which results are shown in table 12. Without further analyses I could summarize that as in my initial evaluations performance of PER and LOC class were reasonable high while performance of ORG class was poor. My model produced a lot of false positives for this class.

- NER model classifies medical abbreviations as ORG entities.
- E.g.: ATC, PTA, AV-Fistel, AEZ, LV ...

Therefore I removed the ORG class from my NE class set to avoid false positives.

6.3.2 Binary PHI Recognition

Table 13 shows the results of my binary PHI evaluation. Using spelling variant detection I improved my baseline F_2-score by 5%. Adding my out of domain NER model raised the F_2-score by another 2%. Figure 5 shows that precision is suffering by 9%. It needs to be mentioned that already my out of domain model

21. `https://github.com/MaviccPRP/ger_ner_evals/`

		Predicted	
		ANON	KEEP
True	ANON	605	75
	KEEP	251	13 203

Table 14: PHI Confusion matrix

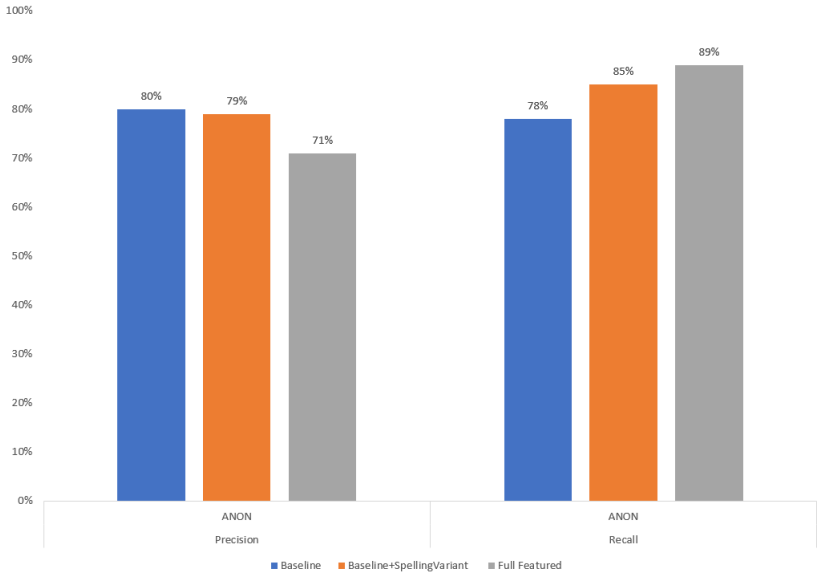

Figure 5: PHI recognition on medical test set

Model	Kappa coefficient
Baseline	0.779
Baseline + Spelling Variant	0.779
Full Featured	0.726

Table 15: Cohen's kappa coefficient for binary PHI recognition

Named Entity	Baseline	Full-Featured
DATE	0.94	0.94
LOC	0.56	0.57
PER	0.41	0.66
PHONE	0.73	0.73
PLZ	0.97	0.97
SALUTE	0.98	0.98
TITLE	0.97	0.97

Table 16: NER on medical test set using F_2-score

lowers precision by 8%, while the spelling variant detection had almost no effect on precision. At the same time recall could be improved by 11%. The baseline model plus spelling variant detection improved recall by 7%, while the full featured model improved it by another 4%. I examined the performance of my full featured model as well by looking at the confusion matrix in table 14. There were 605 correct classified PHI tokens but still I got 251 non-PHI tokens de-identificated and 75 PHI-tokens not de-identificated. This 75 tokens we will target in the next section, as they can potentially contain directly linked PHI tokens like surname and address tokens of patients.

Evaluating the 'quality' of the three classifiers using Cohen's kappa I got the following results listed in table 15. While keeping the coefficient adding my spelling variant detector, the 'quality' of the full featured classifier is slightly suffering, though can be still interpreted as *good* using the interpretation in table 10.

6.3.3 Multiclass NER Evaluation

As the classes PLZ, SALUTE, TITLE, PHONE and DATE are rule-based named entities, there was no performance difference between baseline and the full featured model. Evaluation showed a high F_2-score for purely rule-based classes like PLZ,

	DATE	LOC	O	PER	PHONE	PLZ	SALUTE	TITLE
DATE	225	0	16	0	0	0	0	0
LOC	0	58	38	8	0	0	0	0
O	9	26	13137	284	0	0	0	0
PER	0	4	17	144	0	0	0	0
PHONE	0	0	9	0	15	2	0	0
PLZ	0	0	0	0	0	15	0	0
SALUTE	0	0	1	0	0	0	51	0
TITLE	0	0	2	2	0	0	0	71

Table 17: Multiclass confusion matrix

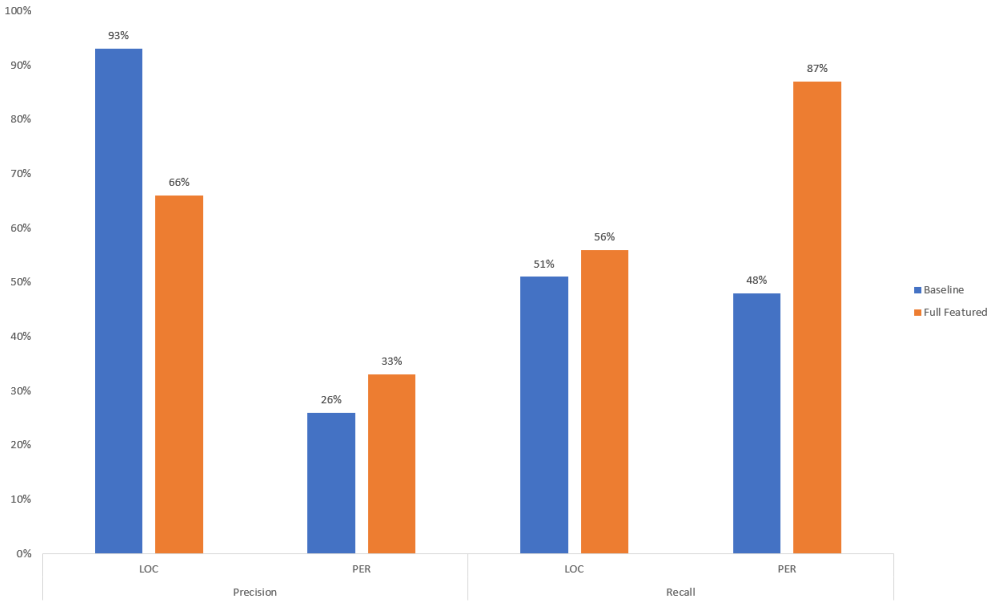

Figure 6: NER on medical test set

Model	Kappa coefficient
Baseline	0.721
Full Featured	0.728

Table 18: Cohen's kappa coefficient for binary PHI recognition

SALUTE, TITLE and DATE. F_2-score for PHONE class was much lower than the other rule-based recognized named entities. The shape of phone numbers was highly variable thus they were hard to specify by a regular expression.

The PER class improved F_2-score by 25% whereas the LOC class made almost no improvement. A closer look at precision and recall in figure 6 showed a huge improvement of recall for PER class by almost 40%, while precision could be improved by 6%. The LOC class lost 27% in precision, gaining only 5% in recall. Due to ambiguities of entries in LOC and PER gazetteers (e.g. Schöneberg, Rostock, etc.) precision of LOC class is suffering.

To get a better understanding of the false negatives, I analyzed them more precisely. From 165 PER tokens there were ten false negatives. These negatives contained two surnames with a high cost and eight one letter abbreviations for first names. From 104 LOC tokens, were 35 false negatives. 29 tokens belonged to the term 'Chest Pain Unit', two tokens were house numbers, 2 tokens belonged to a part of a street name and two tokens were a name of an external hospital.

The F_2-score of the PHONE class was the lowest from the rule-based recognized entities. While precision was 100% recall was 58%. This is why I analyzed the false negatives. From 26 phone number tokens nine tokens were not recognized. All of them represented the last token of a phone number, not matched by the regular expression.

The precision of the PER class is comparably low. Because of that I analyzed the false positives more precisely. As noted in table 17 we got 165 PER entities in the gold standard but the full featured model classified 294 PER tokens. 68 false positives were 'Die'. This token was listed in the gazetteer *German and international first names*. 32 false positives were 'Sehr'. This token was classified as PER by the statistical model. The same goes for the tokens 'Pain' and 'Kollegin'. As with the ORG class there were as well some medical terms classified as PER: 'Karotisdoppler', 'Karotisdoppler', 'Normofrequenter', 'Normofrequenter', 'Sinusrhythmus', 'Aorta', 'Gadolinum', etc.

Evaluating the 'quality' of the two classifiers using Cohen's kappa I got the following results listed in table 18. Both classifiers kept considerable good kappa coefficient using the interpretation of Landis and Koch (Landis and Koch 1977).

6.4 Significance Test

6.4.1 Significance Test Binary Model

I tested significance between the binary baseline model and the binary full featured model using McNemar's test and approximate randomization.

To compute the McNemar's statistics I used the contingency table shown in table 19. To find out if one of the two models classify significantly different I defined a p-value of 0.05 and compute the χ^2 value.

		Full featured	
		correct	wrong
Baseline	correct	13 726	82
	wrong	125	201

Table 19: McNemar contincency table binary

$$\begin{aligned}\chi^2 &= \frac{(|82 - 125| - 1)^2}{82 + 125} \\ &= 8.52173913043\end{aligned}$$

Looking up in the χ^2 table I read a p-value of 0.0035. Thus with a probability of 0.0035 I get such an annotation result like in table 19, if the null hypothesis is true. As my p-value is 0.05 I could reject the null hypothesis and suppose that my two classifiers predicted significantly different. Furthermore I could read in the table that my full featured model had a lower error rate than my baseline model. My full featured model made 82 misclassifications where my baseline model classified correctly. In contrast the full featured model classified 125 times correctly where my baseline model misclassified.

Using approximate randomization I split my test data set into sixteen strata. Each strata represented a single medical admission note. My statistic is F_2-score, that is why I needed to count true positive (TP), false positive (FP) and false negative (FN) instances per strata. With these counts I could produce a triple for each strata containing successful predictions, all predictions of my model and all

Strata No.	TP	TP+FP	TP+FN
1	6	8	7
2	29	40	36
3	24	31	31
4	45	54	61
5	37	53	51
6	32	41	41
7	27	32	30
8	32	39	43
9	35	48	48
10	48	58	59
11	40	45	46
12	20	30	29
13	32	34	40
14	29	35	37
15	44	54	61
16	50	61	60

Table 20: Statistics of my stratified test set for the **baseline**. Column 1 contains strata number, column 2 the correct predictions of my model, column 3 all predictions of my model and column 4 all annotations of the gold standard.

predictions of the gold standard. Table 20 and 21 show the counts for the baseline model and the full featured model.

With an iteration number of 1000 I got a p-value of 0.0009. Supposing a p-value of 0.05 I could assume the two classifiers differ significantly in their predictions using F_2-score.

6.4.2 Significance Test Multiclass Model

To test if my two multiclass models classify significantly different I used McNemar's test. Table 22 shows the contingency table for the comparison of the models. As in the previous McNemar's test I saw that my full featured model had a lower error rate than the baseline model.

Strata No.	TP	TP+FP	TP+FN
1	7	9	7
2	35	56	36
3	29	40	31
4	51	66	61
5	45	75	51
6	41	59	41
7	27	39	30
8	36	51	43
9	39	56	48
10	54	76	59
11	43	51	46
12	26	38	29
13	33	36	40
14	31	46	37
15	54	78	61
16	54	80	60

Table 21: Statistics of my stratified test set for the **full featured model**. Column 1 contains strata number, column 2 the correct predictions of my model, column 3 all predictions of my model and column 4 all annotations of the gold standard.

		Full featured	
		correct	wrong
Baseline	correct	13 643	73
	wrong	106	312

Table 22: McNemar contincency table multiclass

$$\begin{aligned}\chi^2 &= \frac{(|73-106|-1)^2}{73+106} \\ &= 5.72067039106\end{aligned}$$

Looking up in the χ^2 table I read a p-value of 0.017. Thus with a probability of 0.017 I get such a result, if the null hypothesis is true. As my p-value is 0.05 I could reject the null hypothesis and suppose that my two classifiers predict significantly different.

7 Summary and Future Work

Medical texts are a huge resource for computational research. Due to privacy issues they are mostly difficult to access. This is especially true for German medical texts. Though there is a public interest in medical data processing the lack of shared data makes research a hard task. This project had the objective to build a prototype of a stand alone tool for automatic PHI recognition and de-identification of medical admission notes from the cardiology domain.

I had access to a large not annotated medical text corpus containing around 180 000 medical admission notes. Semi-structured headers and a small pre-annotated test set gave me the possibility to design and evaluate a pipeline for automatic binary PHI recognition and multiclass NER using well known rule-based approaches and statistical NER models. I combined a NER model based on regular expressions and gazetteers, a spelling variant detection algorithm based on Levenshtein distance and a statistical NER model trained on out of domain corpora.

Using the rule-based NER model as a baseline for binary PHI detection I showed that the spelling variant detector improved F_2-score significantly from 78% to 85%. While the spelling variant detector improved recall from 78% to 85% precision decreased just by 1%. Adding the statistical model improved recall by another 4% while precision decreased by 8%,

My multiclass NER model gave further insides into the performance of my de-identification algorithm. My statistical model especially improved recall for the highly sensible PER tokens containing first names and last names, from 48% to 87% while precision could be improved on a low level from 26% to 33%. To a lesser extend recall of the LOC class could be improved from 51% to 56%. Still precision of LOC recognition suffered from 93% to 66%. However not de-identifying a PHI token had a higher cost for my task than de-identification of a non-PHI token.

Besides the improvement of PHI and multiclass NER recognition scores the statistical NER step added semantic information to my de-identified data, as additional PHI tokens could be replaced by semantically valuable NE labels. Applying statistical tests using McNemar's test and approximate randomization I could show that my full featured de-identification tool improved performance in binary PHI recognition and multiclass NER significantly.

Because of the lack of published research work on de-identification of German medical texts this project understands itself as a proof of concept. Further research needs to be done to improve recall while keeping precision high. As (Uzuner et al. 2008) integrated a precision improvement step using support vector machines this could be done for my medical set, too. In addition (Wellner et al. 2007) showed that already small annotated training sets can increase recall significantly. This is why I established an annotation project based on WebAnno[22] to encourage authorized medical researchers to annotate training and evaluation data. An encouraging research for the use of smaller training sets was done by (Scheurwegs et al. 2013) who used 100 Dutch medical records for training and (Z. Liu et al. 2017) who trained a bidirectional LSTM model on a training set of 600 annotated mental health records reaching promising results. Using out of domain data GloVe word embeddings trained on German Wikipedia might be a promising approach, as they contain a wider domain than our NER corpora CoNLL, GermEval and Eurpoparl (Pennington, Socher, and Manning 2014; Dernoncourt et al. 2016). In general NER models trained on Wikipedia perform well on narrow domains (Nothman, Murphy, and Curran 2009; Balasuriya et al. 2009).

Next to improving recall and precision my algorithm currently performs slow, as huge gazetteers containing regular expressions demand high computational power. This can be possibly avoided by optimizing the current algorithm using a parallel file handling. Another option would be to replace complex regular expressions for German streets and villages with context information of tokens about house numbers and suffixes.

22. `https://webanno.github.io/webanno/`

References

A Beckwith, Bruce, Rajeshwarri Mahaadevan, Ulysses Balis, and Frank Kuo. 2006. „Development and evaluation of an open source software tool for deidentification of pathology reports“. In *BMC medical informatics and decision making,* 6:12.

Balasuriya, Dominic, Nicky Ringland, Joel Nothman, Tara Murphy, and James R. Curran. 2009. „Named Entity Recognition in Wikipedia“. In *Proceedings of the 2009 Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources,* 10–18. People's Web '09. Suntec, Singapore: Association for Computational Linguistics. `http://dl.acm.org/citation.cfm?id=1699765.1699767`.

Berman, J. 2003. „Concept-match medical data scrubbing. How pathology text can be used in research“. In *Arch Pathol Lab Med.*

Ciaramita, Massimiliano, and Yasemin Altun. 2005. „Named-Entity Recognition in Novel Domains with External Lexical Knowledge“. `http://www.cis.upenn.edu/%5C~%7B%7Dcrammer/workshop%5C_material/ciaramita%5C_altun%5C_structlearn.pdf`.

Dernoncourt, Franck, Ji Young Lee, Özlem Uzuner, and Peter Szolovits. 2016. „De-identification of Patient Notes with Recurrent Neural Networks“. *CoRR* abs/1606.03475. arXiv: `1606.03475`. `http://arxiv.org/abs/1606.03475`.

Douglass, M., G. D. Clifford, A. Reisner, G. B. Moody, and Mark RG. 2004. „Computer-assisted de-identification of free text in the MIMIC II database“. In *Computers in Cardiology, 2004,* 341–344.

Faruqui, Manaal, and Sebastian Padó. 2010. „Training and Evaluating a German Named Entity Recognizer with Semantic Generalization“. In *Proc. of KONVENS 2010.*

Ferrandez, O., B. R. South, S. Shen, F. J. Friedlin, M. H. Samore, and S. M. Meystre. 2013. „BoB, a best-of-breed automated text de-identification system for VHA clinical documents“. *J Am Med Inform Assoc* 20:77–83.

Finkel, Jenny Rose, Trond Grenager, and Christopher Manning. 2005. „Incorporating non-local information into information extraction systems by gibbs sampling“. In *In ACL,* 363–370.

Gardner, James, and Li Xiong. 2008. „HIDE: An integrated system for health information de-identification“. In *In CBMS.*

Guo, Honglei, Huijia Zhu, Zhili Guo, Xiaoxun Zhang, Xian Wu, and Zhong Su. 2009. „Domain Adaptation with Latent Semantic Association for Named Entity Recognition“. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics,* 281–289. NAACL '09. Boulder, Colorado: Association for Computational Linguistics. `http://dl.acm.org/citation.cfm?id=1620754.1620795`.

Gupta, Dilip, Melissa Saul, and John Gilbertson. 2004. „Evaluation of a Deidentification (De-Id) Software Engine to Share Pathology Reports and Clinical Documents for Research“. *American Journal of Clinical Pathology* 121 (2): 176–186. `/oup/backfile/content_public/journal/ajcp/121/2/10.1309/e6k33gbpe5c27fyu/2/ajcpath121-0176.pdf`.

K. Buckland, Michael, and Fredric Gey. 1994. „The Relationship between Recall and Precision.“, 45:12–19.

Kulkarni, Vivek, Yashar Mehdad, and Troy Chevalier. 2016. „Domain Adaptation for Named Entity Recognition in Online Media with Word Embeddings“. *CoRR* abs/1612.00148. arXiv: `1612.00148`. `http://arxiv.org/abs/1612.00148`.

Landis, J. R., and G. G. Koch. 1977. „The measurement of observer agreement for categorical data“. *Biometrics* 33 (1): 159–174.

Lee, Ji Young, Franck Dernoncourt, Özlem Uzuner, and Peter Szolovits. 2016. „Feature-Augmented Neural Networks for Patient Note De-identification“. *Computing Research Repository* abs/1610.09704. arXiv: `1610.09704`. `http://arxiv.org/abs/1610.09704`.

Liu, Z., B. Tang, X. Wang, and Q. Chen. 2017. „De-identification of clinical notes via recurrent neural network and conditional random field“. *J Biomed Inform* 75S:S34–S42.

Liu, Zengjian, Yangxin Chen, Buzhou Tang, Xiaolong Wang, Qingcai Chen, Haodi Li, Jingfeng Wang, Qiwen Deng, and Suisong Zhu. 2015. „Automatic de-identification of electronic medical records using token-level and character-level conditional random fields“. Proceedings of the 2014 i2b2/UTHealth Shared-Tasks and Workshop on Challenges in Natural Language Processing for Clinical Data, *Journal of Biomedical Informatics* 58 (Supplement): S47–S52. `http://www.sciencedirect.com/science/article/pii/S1532046415001197`.

Manning, Christopher D., Mihai Surdeanu, John Bauer, Jenny Finkel, Prismatic Inc, Steven J. Bethard, and David Mcclosky. 2014. „The Stanford CoreNLP Natural Language Processing Toolkit“. In *In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations,* 55–60.

McNemar, Quinn. 1947. „Note on the sampling error of the difference between correlated proportions or percentages“. *Psychometrika* 12 (2): 153–157. `https://doi.org/10.1007/BF02295996`.

Meystre, Stephane M., F. Jeffrey Friedlin, Brett R. South, Shuying Shen, and Matthew H. Samore. 2010. „Automatic de-identification of textual documents in the electronic health record: a review of recent research“. *BMC Medical Research Methodology* 10, number 1 (): 70. `https://doi.org/10.1186/1471-2288-10-70`.

Morgan, William. No date. „Statistical Hypothesis Tests for NLP or: Approximate Randomization for Fun and Profit“. Stanford NLP Group. `http://cs.stanford.edu/people/wmorgan/sigtest.pdf`.

Neamatullah, Ishna, Margaret Douglass, Li-wei H Lehman, Andrew Reisner, Mauricio Villarroel, William Long, Peter Szolovits, George B. Moody, Roger G Mark, and Gari D Clifford. 2008. „Automated De-Identification of Free-Text Medical Records“. *BMC MEDICAL INFORMATICS AND DECISION MAKING* 8 (32).

Noreen. 1988. *Introduction to Testing Hypotheses Using Computer Intensive Methods.* New York, NY, USA: John Wiley & Sons, Inc.

Nothman, Joel, Tara Murphy, and James R. Curran. 2009. „Analysing Wikipedia and Gold-standard Corpora for NER Training". In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics,* 612–620. EACL '09. Athens, Greece: Association for Computational Linguistics. http://dl.acm.org/citation.cfm?id=1609067.1609135.

Padó, Sebastian. 2006. *User's guide to sigf: Significance testing by approximate randomisation.*

Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. 2014. „GloVe: Global Vectors for Word Representation". In *Empirical Methods in Natural Language Processing (EMNLP),* 1532–1543. http://www.aclweb.org/anthology/D14-1162.

Persson, Adam. 2017. *The Effect of Excluding Out of Domain Training Data from Supervised Named-Entity Recognition.* Gothenburg, Sweden. http://www.aclweb.org/anthology/W17-0240.

rbx. 2014. *Cohen's kappa in plain English.* Cross Validated. URL:https://stats.stackexchange.com/q/82187 (version: 2017-10-29). eprint: https://stats.stackexchange.com/q/82187. https://stats.stackexchange.com/q/82187.

Riezler, Stefan, and John T. Maxwell. 2005. „On Some Pitfalls in Automatic Evaluation and Significance Testing for MT". In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization,* 57–64. Ann Arbor, Michigan: Association for Computational Linguistics. http://www.aclweb.org/anthology/W05-0908.

Ruch, P., R. H. Baud, A. M. Rassinoux, P. Bouillon, and G. Robert. 2000. „Medical document anonymization with a semantic lexicon". *Proc AMIA Symp:* 729–733.

Scheurwegs, Elyne, Kim Luyckx, Filip Van der Schueren, and Tim Van den Bulcke. 2013. „De-identification of clinical free text in Dutch with limited training data: A case study“.

Schlunder, I. 2015. „Datenschutzkonforme Lösungen fur die Versorgungsforschung“. In *14. Deutscher Kongress fur Versorgungsforschung.*

Soriano, Ignacio Martinez, and Juan Luis Castro Pena. 2017. „Automatic medical concept extraction from free text clinical reports, a new named entity recognition approach“. *International Journal of Computers.*

Starlinger, Johannes, Madeleine Kittner, Oliver Blankenstein, and Ulf Leser. 2017. „How to improve information extraction from German medical records“. *it - Information Technology* 59 (4): 171.

Stubbs, Amber, Christopher Kotfila, and Ozlem Uzuner. 2015. „Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/UTHealth shared task Track 1“, 58.

Sun, Huiyu, Ralph Grishman, and Yingchao Wang. 2016. „Domain Adaptation with Active Learning for Named Entity Recognition“. In *Cloud Computing and Security: Second International Conference, ICCCS 2016, Nanjing, China, July 29-31, 2016, Revised Selected Papers, Part II,* edited by Xingming Sun, Alex Liu, Han-Chieh Chao, and Elisa Bertino, 611–622. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-48674-1_54.

Sweeney, Latanya. 1996. „Replacing personally-identifying information in medical records, the scrub system“. *Journal of the American Medical Informatics Association.*

———. 2002. „K-anonymity: A Model for Protecting Privacy“. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.* (River Edge, NJ, USA) 10, number 5 (): 557–570. ISSN: 0218-4885. http://dx.doi.org/10.1142/S0218488502001648.

Taira, R. K., A. A. Bui, and H. Kangarloo. 2002. „Identification of patient name references within medical documents using semantic selectional restrictions“. *Proc AMIA Symp:* 757–761.

Thomas, S. M., B. Mamlin, G. Schadow, and C. McDonald. 2002. „A successful technique for removing names in pathology reports using an augmented search and replace method“. *Proc AMIA Symp:* 777–781.

Uzuner, O., T. C. Sibanda, Y. Luo, and P. Szolovits. 2008. „A de-identifier for medical discharge summaries“. *Artificial Intelligence in Medicine* 42 (1): 13–35.

Wagner, Robert A., and Michael J. Fischer. 1974. „The String-to-String Correction Problem“. *J. ACM* (New York, NY, USA) 21, number 1 (): 168–173. `http://doi.acm.org/10.1145/321796.321811`.

Wellner, B., M. Huyck, S. Mardis, J. Aberdeen, A. Morgan, L. Peshkin, A. Yeh, J. Hitzeman, and L. Hirschman. 2007. „Rapidly retargetable approaches to de-identification in medical records“. *J Am Med Inform Assoc* 14 (5): 564–573.

Yeh, Alexander. 2000. „More Accurate Tests for the Statistical Significance of Result Differences“. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 2,* 947–953. COLING '00. Saarbrücken, Germany: Association for Computational Linguistics. `https://doi.org/10.3115/992730.992783`.

Yimam, Seid Muhie, Iryna Gurevych, Richard Eckart de Castilho, and Chris Biemann. 2013. „WebAnno: A Flexible, Web-based and Visually Supported System for Distributed Annotations“. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations,* 1–6. Sofia, Bulgaria: Association for Computational Linguistics. `http://www.aclweb.org/anthology/P13-4001`.

Yuwono, Steven Kester, Hwee Tou Ng, and Kee Yuan Ngiam. 2016. „Automated Anonymization as Spelling Variant Detection“. In *Proceedings of the Clinical Natural Language Processing Workshop (ClinicalNLP),* 99–103. Osaka, Japan: The COLING 2016 Organizing Committee. `http://aclweb.org/anthology/W16-4214`.