Universität Bremen

Anonymizing Personally Identifiable Information with Named-entity Recognition

Bachelor thesis
conducted at Cognitive Systems Lab
Prof. Dr.-Ing. Tanja Schultz
Faculty 3 – Mathematics and Computer Science
University of Bremen

submitted by

Tianyi Wang

Advisor:

Moritz Meier

Supervisors:

Prof. Dr.-Ing. Tanja Schultz
Dr. Felix Putze

Day of registration: 18.08.2021
Day of submission: 04.10.2021

I hereby declare that I have written the present thesis independently, without assistance from external parties and without use of other resources than those indicated.

Bremen, den 04.10.2021

Zusammenfassung

Die Zunahme des Datenbedarfs hat das Forschungsinteresse an der Anonymisierung personenbezogener Daten geweckt, um die Daten nicht mehr persönlich identifizierbar zu machen. Die Named-entity recognition (oder Eigennamenerkennung) ist einer der Ansätze. Durch die Identifizierung von Personennamen und Ortsnamen in Texten ist es möglich, diese sensitiven Informationen zu verschlüsseln und die Daten zu anonymisieren. Bei verschiedenen Experimenten mit unterschiedlichen Algorithmen auf verschiedenen öffentlichen Datensätzen lieferten die trainierten Modelle aus der cross-validation in NLTK und pycrfsuite anständige NER-Ergebnisse, die den Ansatz ermöglichen. Mit den anständigen NER (einige der Modelle überholen das Paper) und Anonymisierung (F1 über 0,9) Ergebnisse. Diese Bachelorarbeit zeigt, dass der Ansatz vielversprechend für die Entwicklung eines NER-basierten Anonymisierungssystems für englische und deutsche Texte ist, und entwickelt ein erstes System für zukünftige Erweiterungen.

Stichwörter: Privacy-preserving natural language processing, Named-entity recognition, NER, NLTK, iterative scaling, conditional random fields, pycrfsuite

Abstract

The rise of data needs has excited the research interest in personal data anonymization to make the data not personally identifiable anymore. Named-entity recognition is one of the approaches. By identifying the person names and locations in texts, it is possible to encrypt this sensitive information to anonymize the data. With various experiments using various algorithms on various public datasets, the trained models from cross-validation in NLTK and pycrfsuite deliver decent results, making the approach possible. With the decent NER (some of the models overperform the paper) and anonymization (F1 over 0.9) results. This thesis shows that the approach holds some promise for developing a NER-based anonymization system for texts in English and German, and develops an initial system for future extensions.

Keywords: Privacy-preserving natural language processing, Named-entity recognition, NER, NLTK, iterative scaling, conditional random fields, pycrfsuite

Contents

List of Figures

List of Tables

1. Introduction

Artificial intelligence (AI)-based approaches are currently dominated by the use of *machine learning* (ML) methods because they can learn and solve a given task independently based on data. However, this poses a major challenge: How can extensive data be bundled and processed while protecting the sensitive content and preserving the privacy of those affected? There can be a risk concerning the privacy of individuals or the sensitive data of the companies because of the high number of possible interactions [GHMAN+10]. Once leaked, the data would be vulnerable to malicious activities.

Personally identifiable information (PII, or personal data) is defined as information that can be used to distinguish or trace an individual's identity either alone or when combined with other public information that can be linked to a specific individual. The growth in identity theft has increased concerns regarding unauthorized disclosure of PII [KW09]. The new *European General Data Protection Regulation* (GDPR[1])) states that explicit consent from the affected individuals is needed to use PII for secondary purposes (different from the primary purpose that motivated the initial collection, such as healthcare or service billing). Ideally, the data collector should strive to gather such consent. However, in practice, this may not be feasible. It may be difficult to contact individuals to gain their consent. Additionally, individuals with rare conditions are more likely to be concerned about their privacy, which makes them less prone to grant consent for their data to be used. These noticeable shortcomings will highly likely lead to biased data.

To avoid the need for consent, data used for secondary purposes should no longer be personally identifiable. Anonymization, also known as *statistical disclosure control* (SDC), provides a way to turn PII into information that cannot be linked to a specifically identified individual anymore and hence is not subject to privacy regulations [HDFSC18].

Since a large part of user data comes from natural language, especially text-based information, I intend to address these tasks in the inter-disciplinary field of privacy-preserved natural language processing. In this case, I will create natural language

[1]EU General Data Protection Regulation, 2016/679. https://gdpr-info.eu

processing (NLP) datasets that preserve user privacy and train machine learning models that store only non-identifying user data. The key method here is PII detection, i.e., how to automatically find those words or phrases in texts that contain PII. Our goal is a function that obtains a piece of input text, classifies its parts that contain PII, and anonymizes them. Its performance must be measurable. It is therefore intended to use the approach of *named-entity recognition* (NER).

NER is a field that has been thriving for more than fifteen years. It aims at extracting and classifying mentions of *rigid designators* (i.e. named-entities) from text. [NS07] NER is widely used in many NLP applications. NER is a source of information for various NLP applications. Moreover, NER is useful in the anonymization of unstructured data. In particular, it can detect those terms that might be used to re-identify an individual and those terms that contain sensitive information [HDFSC18], making it a good fit for our tasks.

To develop such a tool for anonymization one has to understand the data security background, i.e. what PII is, its need to be anonymized, and anonymization. This will be discussed in section 1.1. An overview of the datasets and algorithms shall be thoroughly introduced in chapter 2

1.1 Data security background

1.1.1 Personally identifiable information

There have been various kinds of definitions of personally identifiable information (PII, also known as personal data), preeminently in legal documents. [NS10] The GDPR defines personal data in its article 4[2] as follows:

> Any information relating to an identified or identifiable natural person.

For that matter, some information categories are widely considered as PII. These are, including but are not limited to, name (full name, maiden name), personal identification number (e.g. passport number, social security number), home address, email address, medical records. [KW09] The wide range of PII makes privacy-preserving NLP a promising field that finds its applications in many areas. Since the methods in this thesis are based on NER, the detection of PII should happen on the detection of the PII-relevant NER-tags, i.e. which NER-tags indicate that their tokens contain PII. Therefore, the relevant tags should be determined. This process shall be profoundly discussed in the next chapter. 3.4

1.1.2 Anonymization

Anonymization, also called statistical disclosure control, is a process whereby personal data is transformed into non-personal data (also often referred to as “anonymized data”). SDC methods are used as part of anonymization processes. They attempt to control/limit the risk of re-identification and attribute disclosure through manipulations of the data. [EDF18] This thesis presents a fairly ordinary approach to anonymization. The detected PII will be replaced with its corresponding NER-tag, e.g. "John Smith went to work today." will be converted to "PER PER went to work today." where `PER` stands for "person", a common NER tag.

[2] https://gdpr-info.eu/art-4-gdpr/

1.2 Related work

Besides the ones studied in this thesis, there are quite a few other NER shared tasks and datasets worth reading. However, there are fewer tasks for data in German in comparison to English. There is some work in German data included in the table A.1 next chapter, but not in the thesis, such as Europeana newspapers [Neu16] and DFKI smart data [SMS+18]. Language independence has always been a hotspot in NER. The CoNLL-2002 [TKS02] was a shared task in language-independent NER, whose data was in Dutch and Spanish. Besides CoNLL, there are other major conferences in NER, such as MUC-7 [Chi98] (the sequel of MUC-6 mentioned in chapter 3) and ACE [DMP+04]. There is a paper also focusing on anonymization with NER, which has structured categorization of NER tags and includes coreference resolution in its pipeline. [AAA+19]

1.3 Aim of the thesis

This thesis aims to explore if my proposed NER approach can correctly classify named-entities from some chosen public datasets, and be used for our anonymization purposes. The work of this thesis was planned in roughly 5 iterative stages. To sum up the objectives:

- Build an initial pipeline after initial literature research;
- Expand the pipeline with more data and more algorithms;
- Evaluate the NER and the anonymization performances;
- Compare the performances and establish the conclusion

2. Datasets and algorithms overview

In this chapter, I will introduce the technical background and the building blocks of the machine learning algorithms and libraries, to give the readers a holistic picture of the related research, methodology, and analysis.

Natural language processing is a theoretically motivated range of computational techniques for analyzing and representing naturally occurring texts at one or more levels of linguistic analysis for the purpose of achieving human-like language processing for a range of tasks or applications. [Lid01] Since the 1990s, as the computational power increased [Sch97] and more machine learning methods were employed, the NLP approaches gradually shifted to more statistical ones from the logical (rule-based) ones before. [Kum11] This development is similar to what it has been for one of NLP's subfields, named-entity recognition.

Understandably, there are various entity types in NER. There are the "enamex" (a term from the Sixth Message Understanding Conference, MUC-6 [Sun95]) types, i.e. `person (PER)`, `location (LOC)` and `organization (ORG)`. In some data, the type `miscellaneous (MISC)` is used to note other entity types that are not "enamex". The "timex" (another term from MUC) types `date` and `time` and the "numex" types `money` and `percent` are also quite predominant in other literature. In this thesis, most of the datasets have four entity types, the three "enamex" types and `MISC`. However, certain datasets contain slightly different entity tags. For this thesis, three datasets in English and three datasets in German are used. Since the research project (of which this thesis is a part) of the Cognitive Systems Lab (CSL) is in the beginning phase, the datasets were freely —yet not arbitrarily—chosen from various domains. Four NER training algorithms were used in this thesis. The used datasets and algorithms shall be introduced in the next sections below.

2.1 The used datasets

The dataset study is primarily based on the GitHub repository `entity-recognition-datasets`[1]. The datasets used in the thesis are introduced below. Appendix A can be referred to

[1] https://github.com/juand-r/entity-recognition-datasets

as an overview of all the datasets researched in the thesis.

2.1.1 CoNLL-2003

The CoNLL[2] (The SIGNLL Conference on Computational Natural Language Learning) is a yearly conference organized by SIGNLL (ACL's Special Interest Group on Natural Language Learning). The focus of CoNLL is on theoretically, cognitively, and scientifically motivated approaches to computational linguistics, rather than on work driven by particular engineering applications.

The CoNLL-2003 [SM03] was a shared task aiming at language-independent named-entity recognition. The training and test data offered at this conference were in English and German, which makes the CoNLL-2003 dataset language-wise a perfect fit for this thesis. For each of the languages, there was an official split of the data, i.e. a training data file, a development data file, and two test data files (testa and testb), although all of the data is later merged for cross-validation in the section 3.2.2 in the next chapter. The content of this dataset is from the news. The English data was taken from the Reuters Corpus. This corpus consists of Reuters news stories between August 1996 and August 1997. The text for the German data was taken from the German language part of the ECI Multilingual Text Corpus, which was originally extracted from Frankfurter Rundschau, a German newspaper. The data is in the conventional CoNLL format. All data files contain one word per line with empty lines representing sentence boundaries. Each line of the English data contains four fields: the word, its part-of-speech tag, its chunk tag—like `Noun Phrase (NP)`, `Verb Phrase (VP)`—and its named entity tag, whereas the German data has an additional field standing between the word and the pos tag, the lemma of the word. The NER tag is at the end of each line. It states whether the current word is inside a named entity or not. The tag also encodes the type of the named entity. In the case of CoNLL-2003, there are four entity types, `person (PER)`, `location (LOC)`, `organization (ORG)` and `miscellaneous` NEs `(MISC)`. As was introduced above, the first three entity types are also known as the "enamex". The tagging scheme is the so-called IOB scheme: Assuming that named entities are non-recursive and non-overlapping, if a word (token) is the *beginning* word of an NE, then its NER tag is B-(entity type), such as `B-PER`; if a word (token) is *inside* an NE, then its NER tag is I-(entity type), such as `I-ORG`; if a word is not a (part of) named entity, i.e. it is *outside* of one, then its NER tag is `O`.

Here is an example sentence from the CoNLL-2003 English data:

```
U.N. NNP I-NP I-ORG
official NN I-NP O
Ekeus NNP I-NP I-PER
heads VBZ I-VP O
for IN I-PP O
Baghdad NNP I-NP I-LOC
. . O O
```

Here is an example sentence from the CoNLL-2003 German data:

[2]https://www.conll.org/

```
Das d ART I-NC O
Krokodil Krokodil NN I-NC O
steckt stecken VVFIN I-VC O
seinen sein PPOSAT I-NC O
Kopf Kopf NN I-NC O
in in APPR I-PC O
ein ein ART I-NC O
Zimmer Zimmer NN I-NC O
. . $. O O
```

2.1.2 SEC-filings

The work of SEC-filings [SAVB15] explores a machine learning approach for information extraction of credit risk attributes from financial documents, modeling the task as a NER problem. Being extracted and annotated from eight publicly available financial agreements, the dataset is one of few public datasets in the financial domain. There was also an official split. Five of them were used for training, and three of them were used for the test. Anonymization may be effectively deployed in this domain due to its security criticalness. Therefore, it is meaningful to include a dataset in this domain in this thesis. The data format of SEC-filings is almost identical to CoNLL-2003 2.1.1— four columns for each line except for those empty lines as sentences separators, i.e. the word, the pos tag, the chunk tag, and the NE tag -also the four types from CoNLL—in IOB encoding, although it seems that all of the chunk tags are empty and written as a hyphen(`-`), arguably as a place holder.

2.1.3 WNUT17

The WNUT[3] (Workshop on Noisy User-generated Text) workshop focuses on Natural Language Processing applied to noisy user-generated text, such as that found in social media, online reviews, crowdsourced data, web forums, clinical records, and language learner essays. New words, abbreviations, and expressions emerge incredibly fast on social media, and people tend to use a more informal tone. This makes data from social media quite noisy. The WNUT17 [DNvEL17] shared task aimed at tackling rare and emerging named entities from social media. The data is mainly from Twitter, a major social media platform with rich noisy user-generated data. Additionally, Comments from other platforms like Reddit, Youtube, and StackExchange are also included in the development and test data. The format of this data is noticeably different from the format of the aforementioned datasets. Each line contains only two fields, the word and the NE tag, split by a tabulator character instead of a space character, where the latter seems more common in conventional CoNLL-format. The format of this data is comparable to the tab-separated (TSV) format. There are also more entity types. Besides `person` and `location`, there are `corporation`, `product` (tangible goods, or well-defined services), `creative-work` (song, movie, book, and so on) and `group` (subsuming music band, sports team, and non-corporate organizations). The difference between corporations and groups should especially be noted. Interestingly, the results that the participating teams obtained were very low. The best F1 score from the task was 41.86, much lower than other NER tasks such as CoNLL, which concludes that the recognition of emerging and rare entities remains a big challenge.

[3] http://noisy-text.github.io/2021/

2.1.4 GermEval2014

The GermEval 2014 NER Shared Task[4] is an event that makes available CC-licensed German data with NER annotation with the goal of significantly advancing the state of the art in German NER and pushing the field of NER towards nested representations of named entities. The data was sampled from German Wikipedia and News Corpora. [BBKP14] There is an official split of the data in development, training, and test sets. With the GermEval 2014 being the only well-known shared task for German NER after CoNLL-20032.1.1 over ten years before, the German NER has not been as thoroughly researched compared to English NER. It is argued that linguistic features play an important role. For instance, in English, capitalization is an important feature in detecting NEs. In contrast, German capitalizes not only proper names but all nouns, which makes the capitalization feature much less informative. At the same time, adjectives derived from NEs, which arguably count as NEs themselves, such as *englisch* ("English"), are not capitalized in German, in line with "normal" adjectives. Finally, a challenge in German is compounding, which allows concatenating named entities and common nouns into single-token compounds. Taking these features into consideration, two substantial extensions were introduced into the shared task [BBKP14] compared to CoNLL-2003:

- **Fine-grained labels.** Because of the last one of the aforementioned feature, i.e. the "productive morphology" of the German language, two kinds of new fine-grained labels are introduced. Additional to the traditional IOB-format, `-deriv` marks derivations from NEs such as the previously mentioned *englisch* ("English"), and `-part` marks compounds including a NE as a subsequence *deutschlandweit* ("Germany-wide"). These with `-deriv` and `-part` marked entities were all classified under `MISC` in the CoNLL-2003 German data. [SM03]

- **Embedded markables.** Besides the first-level label—the NER tag as it is in the datasets above, the annotation also includes a column for the second-level label. Instead oversimplifying by assuming all of the NEs are "flat" which was the case in CoNLL-2003 from section 2.1.1, the second-level serves as a tag for (first-level) entities which contain an entity inside of it. For example, [BBKP14] the noun phrase *Technische Universitat Darmstadt* ("Technical University of Darmstadt") denotes an `organization`, whilst *Darmstadt* denotes a `location` (the city in which the university is).

There are some more differences for the data format of GermEval 2014: Instead of *MISC*, it uses `OTH` (other) to describe the entity types that are not "enamex", although there is no significant semantic difference; In addition to an empty line, there is a "meta" line for each sentence. It consists of a hashtag "#", a link where the sentence is extracted from, and the date when it was retrieved.

To summarize, there are four columns for each line, the token index within the sentence, the word (token), the first-level NER tag, and the second-level NER tag. The NER tags have characteristics that are introduced above. All of the columns are separated by tabular characters because the dataset is in TSV format. Here is an example sentence:

[4] https://sites.google.com/site/germeval2014ner/

#	http://de.wikipedia.org/wiki/Manfred_Korfmann		
1	Aufgrund	O	O
2	seiner	O	O
3	Initiative	O	O
4	fand	O	O
5	2001/2002	O	O
6	in	O	O
7	Stuttgart	B-LOC	O
8	,	O	O
9	Braunschweig	B-LOC	O
10	und	O	O
11	Bonn	B-LOC	O
12	eine	O	O
13	große	O	O
14	und	O	O
15	publizistisch	O	O
16	vielbeachtete	O	O
17	Troia-Ausstellung	B-LOCpart	O
18	statt	O	O
19	,	O	O
20	„	O	O
21	Troia	B-OTH	B-LOC
22	-	I-OTH	O
23	Traum	I-OTH	O
24	und	I-OTH	O
25	Wirklichkeit	I-OTH	O
26	”	O	O
27	.	O	O

Table 2: Data format illustration. The example sentence contains five named entities: the locations “Stuttgart”, “Braunschweig” and “Bonn”, the noun including a location part “Troia”-Ausstellung, and the title of the event, “Troia - Traum und Wirklichkeit”, which contains the embedded location “Troia”. (Benikova et al., 2014)

Figure 2.1: An example sentence of GermEval 2014, which was originally Table 2 in the paper [BBKP14]

2.1.5 LegalER

The LegalER [LRMS19] is an NER approach in legal documents in German. The dataset is made up of German court decisions. The NER pipeline of this data is equipped with a holistic fine-grained tagging system to serve its domain. On the first level, there are coarse-grained classes (entity types): `person`, `location`, `organization`, `legal norm`, `case-by-case regulation`, `court decision`, and `legal literature`. On the second level, `person` is subcategorized into `person`, `judge` and `lawyer`; `location` is subcategorized into `country`, `city`, `street` and `landscape`; `organization` is subcategorized into `company`, `institution`, `court`, `brand`; for `legal norm`, `law`, `ordinance` and `European legal norm`; and for `case-by-case regulation`, the subclasses are `regulation` and `contract`. There are no further fine-grained subclasses for the last two coarse-grained entity types. This data also has two columns only, the word (token) and the NE tag. The dataset consists of seven data files, six of them consist of 107 documents and 1 one them consists of 108, although there is no noticeable document separator.

2.2 The used algorithms

The first three algorithms used in the experiments of this thesis were conducted in `NLTK`, a Python library for many common NLP tasks. [BKL09] The last algorithm is

from the library `pycrfsuite` to use the algorithm conditional random fields.

2.2.1 Generalized iterative scaling and improved iterative scaling

Generalized iterative scaling (GIS) [BDPDP96] and Improved iterative scaling (IIS) [Ber97] are two early algorithms in their field. Their purpose was to fit log-linear models. [DR72] A notable example is the multinomial logistic regression (maximum entropy) model. They are widely applied, especially for computing maximum-likelihood estimates of the parameters of exponential models. IIS is supposed to be an improvement of GIS. Unfortunately, these two algorithms were not mathematically distinguished in their introduction papers. To my best knowledge, no literature did so, either.

2.2.2 *Megam*

Megam (MEGA Model Optimization Package) [Dau04] is a software that implements maximum likelihood and maximum a posterior optimization of the parameters of maximum entropy models. To specify, it is an implementation for conjugate gradient ascent (for binary problems) and limited memory BFGS (for multiclass problems). It is claimed that this implementation has surpassed traditional algorithms like GIS and IIS.

2.2.3 Conditional random fields

Conditional random fields are undirected statistical models often applied in machine learning and sequence analysis. It is defined as follows: [LMP01]

Let X be a random variable over data sequences to be labeled, and Y is a random variable over corresponding label sequences. All components Y_i of Y are assumed to range over a finite label alphabet Υ. For example, X might range over natural language sentences and Y range over part-of-speech taggings of those sentences, with Υ the set of possible part-of-speech tags.

Definition 2.2.1 (Conditional random fields)**.** Let $G = (V, E)$ be a graph such that:

$$Y = (Y_v)_{v \in V}$$

So that Y is indexed by the vertices of G. Then (X, Y) is a conditional random field in case, when conditioned on X, the random variables Y_v obey the Markov property with respect to the graph:

$$p(Y_v|X, Y_w, w \neq v) = p(Yv|X, Yw, w \sim v),$$

where $w \sim v$ means that w and v are neighbors in G.

It has been proved that conditional random fields can be applied to part-of-speech tagging [LMP01] and named-entity recognition [Set04]. Moreover, a system based on CRF claimed the state-of-the-art in NER in 2016. [LBS+16]

2.3 Anonymization data: CoNLL-2003 testb

According to the original plan, The ILSE dataset [MM00] should have been used in this thesis to evaluate the anonymization performance. However, due to some misunderstandings in between, ILSE could not be used for this purpose. As it was mentioned in section 2.1.1, there are two test data files in CoNLL-2003. In my NER experiments, only the development, training, and testa are used. The unused testb file is then used for anonymization. Details will be reported in section 3.4.

3. Methods

In this chapter, I will describe the overall data (pre-)processing workflow of the experiments and their setups. Since the libraries used for GIS, IIS, *Megam* and the ones for CRF are fundamentally different, their working processes shall be discussed respectively in section 3.2 and 3.3

3.1 Early work

At first, the built-in entity recognizer of the Python library SpaCy[HMVLB20] was briefly used. However, despite painstaking work to adjust the data format to its request, it failed to deliver satisfying training and evaluation results on the dataset WNUT17 [DNvEL17]. The results were drastically low compared with the paper. Therefore, to my best knowledge, the built-in entity recognizer of SpaCy is not suitable for this task. However, SpaCy was used later in this thesis to assist the experiments on conditional random fields. Details to that are in the section 3.3

3.2 Algorithms used in NLTK

Figure 3.1 shows a typical NER workflow.

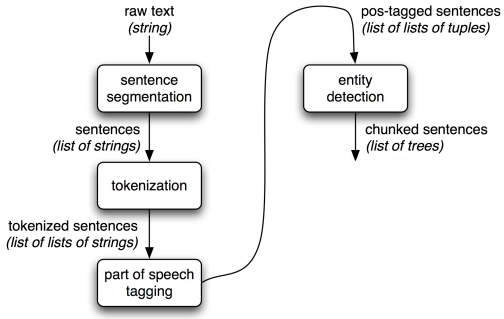

Figure 3.1: NLTK NER workflow from the NLTK book. [BKL09]. One component (relation extraction) from the original figure is removed because it is not relevant for this thesis.

3.2.1 Preprocessing

After the building of an initial pipeline, the official workflow begins with the data preprocessing. Since the initial pipeline was built on the dataset CoNLL-2002 [TKS02] and most of the found datasets are roughly in CoNLL-format from section 2.1.1, it was made sure that the datasets were in or converted into this format so that they can be read by the `ConllCorpusReader` of NLTK. Furthermore, if the data is not integrated into one file (e.g. when there are official splits as mentioned in chapter 2), it should be merged into one file, then split by document, or by sentence if the information that is necessary for a document-level split does not exist in the dataset. Then the data shall be ready for cross-validation.

All the data need to be merged and split. For the data in (almost) exact CoNLL-format, which can be directly read by the `ConllCorpusReader`. In this case, only the merging and document-level splitting are necessary. These datasets are CoNLL-2003 (German and English), sec-filings. Nevertheless, according to the data format, there have been additional preprocessing steps in between:

- The data without part-of-speech tags. These datasets are GermEval 2014, LegalER, and WNUT17. An additional column of the pos tag has to be added because it is not permitted for the `ConllCorpusReader` to read data without pos-tags. It was assumed that a proper pos-tagging would not be necessary because the pos-tags are assumably not required for the actual NER training, since the data does not contain them at all. Therefore, All of the tokens were arbitrarily assigned the pos-tag `UNK` (unknown). This tag is also used by the `tagset_mapping` of NLTK for unrecognized tags. [BKL09]
- In the cross-validation, it was reported that NLTK could not read the WNUT17 data, because there was a hyphen in the NE tag `creative-work`, so the hyphen was removed (`creativework`). This should not affect the performance, since it still exists.
- It should also be noted that the document-level splitting is not always an option, although it seems the most meaningful solution because of the contexts from which NER may benefit. For WNUT17 and GermEval 2014, there are only sentence-level splits in the data. However, since the sentences were originally separately extracted from different documents, the splits can be regarded as document-level splits. For LegalER, like it was mentioned in section 2.1.5, there were over 100 documents in each data file, [LRMS19] but there were no document separating characters whatsoever. As a result, only sentence-level splitting was possible for this data.

3.2.2 Cross-validation

Despite the integrated cross-validation function of the popular Python library scikit-learn[1], I implemented here a cross-validation function myself, because the function from scikit-learn does not quite fit in this task due to various reasons such as the data types. The CV function was kept as "automated" as possible. Five parameters will be passed into the function:

[1] https://scikit-learn.org/stable/modules/cross_validation.html

- `k`. The integer k as in k-fold cross-validation. `k=10` has been a golden standard for all the experiments except for SEC-filings, which only has 8 documents, making it only possible for an 8-fold document-level cross-validation.

- `language`. A string. The function recognizes only the strings `deu`and `eng`, which represents German and English languages. The intention of this parameter was the difference in the constructor call of `ConllCorpusReader`.

- `dataform`. A string. The function recognizes the strings `conll`, `conllpos` and `germeval`. This parameter was defined due to the different (preprocessed) data format. It was explicitly asserted that the combination of `language` and `dataform` were correct.

- `datadir`. A string. The directory in which the preprocessed data is.

- `resultsdir`. A string. The directory to which the results are exported through pandas[2].

The function will be called with these parameters. The split data shall be shuffled, and stored to the corresponding folds. The constructor of `ConllCorpusReader` will be properly called according to the data format and the language of the data, then a model will be trained on the training data formed by the training folds, and evaluated on the evaluation fold. After all the iterations of the cross-validation are finished, the results determined by the built-in `evaluate` function of NLTK are stored in the pandas `DataFrame`, which is exported in a CSV file into the results directory.

To test other algorithms other than *megam*, I overrode the constructor of the `_classifier_builder`, to change the parameter `algorithm` to "GIS" and "IIS" for their experiments.

3.3 Conditional random fields

The CRF approach is based on the code from the contribution of the GitHub repository CRF_NER. [Mas20]

3.3.1 Preprocessing

Before being trained, the data has to be tokenized. In this case, the library SpaCy [HMVLB20] is deployed. The data is preprocessed into SpaCy data format—a `list` of `tuples` in which from one sentence every token and its NE tag (label) are—for tokenizing. Since SpaCy does not take pos-tag as a mandatory argument, only this information has to be loaded. Therefore, it should be specified that which columns of the datasets should be loaded, because the word and the NE-tag are on different positions of the data. Once properly extracted, the data format is ready for cross-validation.

[2]https://pandas.pydata.org/

3.3.2 Cross-validation

The design concept of my CV implementation for CRF is based on the implementation for NLTK above in section 3.2.2. Using the `sent2features` function, the necessary features of the words are extracted from the sentence, then the tagger from the library `pycrfsuite` calls the CRF algorithm for training and predictions. Lastly, the `metrics` module from `scikit-learn` is used for metrics calculations.

3.4 Anonymization

For the anonymization, an "inter-dataset" model should be trained to get better NER results, hence better anonymization results. Since the data domain of WNUT17 (2.1.3) and LegalER (2.1.5) are too specific, which may lead to low performance when anonymizing data from other domains, only CoNLL-2003 English and SEC-filings are used to train the final English model, and CoNLL-2003 German and GermEval 2014 for the final German model. However, since we only have the data for anonymization in German, only the German model can be evaluated. The training algorithm is slightly different from the cross-validation algorithms, but essentially no different. For the anonymization, there are three final models tested. As there are no pre-developed functions to calculate anonymization results, a hand-made algorithm is developed.

1. The model trained out of CoNLL-2003 German data on *Megam*, since it has the best recall score (see section 4.1) among non-CRF algorithms. The recall is more important in anonymization because of its security "criticalness".
2. The model trained out of the joint data of CoNLL-2003 German and GermEval 2014 on CRF.
3. The model trained out of CoNLL-2003 German data on CRF.

4. Results

In this section, I will present the results of the performance of the cross-validation experiments of NER and the anonymization performance of the final German models.

4.1 NER results

All the percentage numbers here are the mean of the cross-validation results. The true positives (TP) or false positives (FP) are the predicted positives (e.g. the model classifying a token as `PER`) that are correct or incorrect. The true negatives (TN) or false negatives (FN) or the predicted negatives (e.g. the model classifying a token as `O`) that are correct or incorrect.[1] Combining these numbers, a confusion matrix can be plotted, and the other metrics can be calculated as shown below.

Definition 4.1.1 (Accuracy)**.**

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Definition 4.1.2 (Precision)**.**

$$Precision = \frac{TP}{TP + FP}$$

Definition 4.1.3 (Recall)**.**

$$Recall = \frac{TP}{TP + FN}$$

Definition 4.1.4 (F1-score)**.**

$$f1 = 2 \times \frac{precision \times recall}{precision + recall}$$

[1] https://towardsdatascience.com/confusion-matrix-for-your-multi-class-machine-learning-model-ff9aa3bf7826

4.2 Results of GIS, IIS, *Megam*

There should originally be in total 18 result files to report because there are three algorithms and six datasets. However, it was reported that the combination of the dataset LegalER and the algorithm *megam* was not possible, because the merged data file is too big for the configuration of *megam* (24MB). Therefore, there are in total 17 result files, which are shown in the following tables. 4.1 shows the accuracies of each algorithm on each dataset. 4.2 shows the precision, 4.3 shows the recall, and 4.4 shows the f-measure. There is an answer[2] on StackOverflow that explains the inconsistency of denominators in NLTK while calculating the metrics.

	CoNLL2003 (Ger)	LegalER	GermEval2014	CoNLL2003 (Eng)	SEC-filings	WNUT17
GIS	0.96	0.95	0.94	0.96	0.98	0.95
IIS	0.95	0.95	0.94	0.96	0.98	0.95
Megam	0.95	*No results*	0.93	0.95	0.98	0.95

Table 4.1: IOB Accuracy by datasets and algorithms

	CoNLL2003 (Ger)	LegalER	GermEval2014	CoNLL2003 (Eng)	SEC-filings	WNUT17
GIS	0.80	0.67	0.77	0.86	0.89	0.68
IIS	0.77	0.57	0.76	0.82	0.89	0.69
Megam	0.70	*No results*	0.71	0.73	0.89	0.62

Table 4.2: Precision by datasets and algorithms

	CoNLL2003 (Ger)	LegalER	GermEval2014	CoNLL2003 (Eng)	SEC-filings	WNUT17
GIS	0.51	0.66	0.40	0.81	0.60	0.16
IIS	0.50	0.60	0.37	0.79	0.57	0.14
Megam	0.53	*No results*	0.30	0.73	0.65	0.17

Table 4.3: Recall by datasets and algorithms

	CoNLL2003 (Ger)	LegalER	GermEval2014	CoNLL2003 (Eng)	SEC-filings	WNUT17
GIS	0.62	0.67	0.52	0.83	0.70	0.26
IIS	0.60	0.58	0.50	0.81	0.68	0.23
Megam	0.60	*No results*	0.42	0.73	0.73	0.27

Table 4.4: F-Measure by datasets and algorithms

There were no systematic recordings of the executing time of the cross-validation processes, because the user scenario of the task is more security-critical than time-critical. However, it was noticeable that the training of LegalER had taken very much time. It took 5 days and 18 hours for GIS and 5 days 11 hours for IIS.

In the exported CSV files of the results, the confusion matrix entries (true positives, true negatives, and false negatives) were also exported. Nevertheless, it would be less meaningful to include these metrics in the tables, considering that they are not as intuitive as the other metrics, and precision, recall, and F-measure are already included.

[2]https://stackoverflow.com/questions/17325554/difference-between-iob-accuracy-and-precision, the answer of alvas. To my best knowledge, there is no such explanation in the NLTK book, documentation, or any other scientific literature whatsoever.

4.3 Results of CRF

Both the weighted and macro results of the precision, recall and F-score were calculated. I decided to only report the macro ones, because it is by definition closer to our goal and arguably the calculation methods of NLTK. Both 8-fold and 10-fold CVs were experimented on SEC-filings for variable control. It should also be noted that the experiments on LegalER were on the fine-grained instead of the coarse-grained version of data.

	Accuracy	Precision	Recall	F-Score
CoNLL2003 German	0.97	0.66	0.56	0.60
LegalER fine-grained	0.99	0.88	0.74	0.80
GermEval2014	0.96	0.54	0.39	0.43
CoNLL2003 English	0.98	0.87	0.77	0.80
SEC-filings 10-fold	0.99	0.92	0.83	0.86
SEC-filings 8-fold	0.99	0.91	0.84	0.86
WNUT17	0.95	0.55	0.31	0.38

Table 4.5: CRF metrics

4.4 Anonymization results

For the NE tags `PER`, `LOC`, `ORG` and `MISC`, I chose `PER` and `LOC` as PII-relevant tags for the semantic meaningfulness. This turns the anonymization problem to a binary-class problem from a multiclass NER problem, i.e. even if a `PER` is classified as `LOC`, it still counts as true positive, because it can still be anonymized. Correspondingly, if a `ORG` is classified as `O`, it is still a true negative.

	CoNLL, Megam	CoNLL+GermEval, CRF	CoNLL, CRF
Precision	0.74	0.91	0.95
Recall	0.81	0.93	0.95
F-Score	0.77	0.92	0.95

Table 4.6: Anonymization results

5. Discussions

In this chapter, I will compare the results to the ones from the papers, and add some more remarks.

5.1 Comparison with papers

The CoNLL-2003 paper [SM03] did not record the accuracy. The results of the models from NLTK at CoNLL-2003 German are at the lower half of the table, which consists of sixteen system results from the conference. The table is sorted by F-score. On the one hand, the precision has been very high. The GIS precision is even at approximately the second or third place of the table for both English and German data. On the other hand, the recall on CoNLL has been relatively low. The recall on the German data is at about the fourth from the bottom, English fifth. The performances of CRF are also about at the lower half of the table.

Unfortunately, the performance on LegalER is significantly lower than it is from the paper. [LRMS19] A possible reason is the data format and preprocessing. The fine-grained labels were just brutally replaced with the coarse-grained labels by text functions in Python. The paper used other methods like conditional random fields and BiLSTM, which could support such fine-grained and coarse-grained recognition, which -to my best knowledge- is not supported in NLTK. In comparison, CRF delivers significantly better results, though still lower than the customized CRF and BiLSTM models from the paper.

In GermEval 2014 [BBKP14], I referred to the Table 7, which is the table for their Metric 2, which collapsed the sub-entity types. This method is the closest to what was done in this thesis. The situation is proportionally similar as it is for CoNLL-2003. High precisions and low recalls let the f-score make to the lower half of the table. The situation is similar for the CRF models as well.

For SEC-filings. the experiments conducted in the thesis are closest to Experiment3 in Table 2 of the paper. [SAVB15] The precision is 89%, slightly lower than the paper's 94, but the best recall is 65%, worse than the paper's 77%. The F score is lower than the paper by 10 percent. For this dataset, CRF even delivers better

results than the paper which does not apply cross-validation. The recall is higher by 10 percent, making the F-score better than the paper by circa 3%.

The WNUT paper [DNvEL17] only recorded the F-score. There is the F1 (entity) and the F1 (surface). Based on the definitions, the F1 (entity) corresponds to the calculation method of the F score in this thesis. The performance of NLTK is slightly above the lowest in the paper. The results of CRF are much more decent, being about the fourth in the table, only 3 percent lower than the best result.

5.2 More remarks and Summary

It was noticed that the recall was significantly lower than precision in many experiments on German data. Admittedly, this was always the case in many papers, such as the German NER in CoNLL-2003 [SM03] and GermEval 2014 [BBKP14]. This is possibly due to the linguistic features of the German language. As it was described in introduction2.1.4, the features such as the capitalization of words may lead to "confusion" of the model, that some entities may be missed. The overall performance of German NER is lower than English NER. There is still a long way to go.

As opposed to the imagination before the experiments, the GIS algorithm performed -in most cases- better than IIS, and *Megam*. This is a surprise because *Megam*was originally the default algorithm when the constructor of `_classifier_builder` in the NLTK NER was called, so the constructor had to be overwritten to try other algorithms.

Though being famous as the then state-of-the-art NER algorithm, CRF has not been predominant in every direction. However, for the tasks in which NLTK significantly underperform, CRF seems to be able to deliver much better results, e.g. LegalER and WNUT17. These two datasets share similar formats, i.e. every line consists of only word and NE tag, and uncommon NE sets, especially with the fine-grained and coarse-grained system in LegalER (details see 3.2.1). Possibly, the data format is the obstacle and the preprocessing was not sufficient to bridge it. These two kinds of data may be "too much" for NLTK, but not for CRF. Following this pattern, it may be concluded that CRF thrives in domain-specific data. Another advantage of CRF is the execution time. Not all the experiments were logged with the executing time, therefore this was not reported. But with the existing log and the personal memory, it can be told that the CRF uses noticeably less time for cross-validations.

However, it can be concluded that the models & algorithms used in the thesis are domain-independent, because they can deliver decent results even from specific domains.

5.3 Anonymization

Although CRF was not as predominant in NER, it is proved to be much better in anonymization, especially with the high recall, which is attached great importance in such a task.

5.4 Future work

As NLTK algorithms and CRF tend to compensate each other, it can be advisable to combine these for future system developments instead of simply replacing them. Moreover, Future work using more algorithms other than the introduced ones above will be a great extension to this work. Moreover, there are other available models, such as the ACE Named Entity Chunker (maximum entropy). It is integrated into NLTK under the id `maxent_ne_chunker` and can be used for English NER.

The data search conducted in this thesis has been quite thorough, especially the German data, considering public datasets for German NER are not as many. In the future, More datasets on the table A.1 could be researched and fed to the models. Especially, if there is a good licensing opportunity, the MUC datasets [Sun95] [Chi98] are also solid NER resources, considering their licenses are quite expensive.[1]

Due to time reasons, I only overrode a small component of NLTK in the work. In the future, more parts of NLTK could be overridden to optimize the workflow and improve performance. For instance, the `ConllCorpusReader` of NLTK only reads data files from the disk. If it can read files from Python data structures as well, the computing time of the workflow could considerably be reduced without the repeating reading-ins and writing-backs.

For anonymization purposes, it would be advisable to investigate NER performances for only the NE-tags that are relevant for the anonymization to compensate for the self-defined anonymization metrics introduced in section 4.4. In this thesis, these are *PERSON* and *LOCATION*.

Since the primary purpose of this thesis was NER, the anonymization method was crudely developed. In further research, Cryptographical encryption methods can be deployed for better anonymization.

[1] MUC-6: https://catalog.ldc.upenn.edu/LDC2003T13; MUC-7: https://catalog.ldc.upenn.edu/LDC2001T02

6. Conclusions

This thesis is an initial attempt to investigate if it is possible to anonymize personal data through named-entity recognition. Personal data, such as person names and locations, can be identified using properly trained NER models, and anonymized with minor encryption. The aim of this study was to build a pipeline for that matter, and expand the system iteratively as much as possible. With the results we have now, it can be concluded that the approach works. The thesis was conducted at the Cognitive Systems Lab at the University of Bremen and all the data used in this thesis is public.

Bibliography

[AAA+19] Allison Adams, Eric Aili, Daniel Aioanei, Rebecca Jonsson, Lina Mickelsson, Dagmar Mikmekova, Fred Roberts, Javier Fernandez Valencia, and Roger Wechsler. AnonyMate: A toolkit for anonymizing unstructured chat data. In *Proceedings of the Workshop on NLP and Pseudonymisation*, pages 1–7, Turku, Finland, September 2019. Linköping Electronic Press.

[BBKP14] Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Padó. GermEval 2014 Named Entity Recognition Shared Task: Companion Paper. In *Proceedings of the KONVENS GermEval workshop*, pages 104–112, Hildesheim, Germany, 2014.

[BDPDP96] Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71, 1996.

[Ber97] Adam Berger. The improved iterative scaling algorithm: A gentle introduction, 1997.

[BKL09] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O'Reilly Media, 2009.

[Chi98] Nancy A. Chinchor. Overview of muc-7/met-2. 1998.

[Dau04] Hal Daumé III. Notes on CG and LM-BFGS optimization of logistic regression. Paper available at http://pub.hal3.name#daume04cg-bfgs, implementation available at http://hal3.name/megam/, August 2004.

[DMP+04] G. Doddington, A. Mitchell, Mark A. Przybocki, L. Ramshaw, S. Strassel, and R. Weischedel. The automatic content extraction (ace) program - tasks, data, and evaluation. In *LREC*, 2004.

[DNvEL17] Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[DR72] J. N. Darroch and D. Ratcliff. Generalized Iterative Scaling for Log-Linear Models. *The Annals of Mathematical Statistics*, 43(5):1470 – 1480, 1972.

[EDF18] Mark Elliot and Josep Domingo-Ferrer. The future of statistical disclosure control, 2018.

[GHMAN+10] José María Gómez-Hidalgo, José Miguel Martín-Abreu, Javier Nieves, Igor Santos, Felix Brezo, and Pablo G. Bringas. Data leak prevention through named entity recognition. In *2010 IEEE Second International Conference on Social Computing*, pages 1129–1134, 2010.

[HDFSC18] Fadi Hassan, Josep Domingo-Ferrer, and Jordi Soria-Comas. Anonymization of unstructured data via named-entity recognition. In Vicenç Torra, Yasuo Narukawa, Isabel Aguiló, and Manuel González-Hidalgo, editors, *Modeling Decisions for Artificial Intelligence*, pages 296–305, Cham, 2018. Springer International Publishing.

[HMVLB20] Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. spaCy: Industrial-strength Natural Language Processing in Python, 2020.

[Kum11] Ela Kumar. *Natural language processing*. IK International Pvt Ltd, 2011.

[KW09] Balachander Krishnamurthy and Craig E. Wills. On the leakage of personally identifiable information via online social networks. In *Proceedings of the 2nd ACM Workshop on Online Social Networks*, WOSN '09, page 7–12, New York, NY, USA, 2009. Association for Computing Machinery.

[LBS+16] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *CoRR*, abs/1603.01360, 2016.

[Lid01] Elizabeth D Liddy. Natural language processing. 2001.

[LMP01] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[LRMS19] Elena Leitner, Georg Rehm, and Julian Moreno-Schneider. Fine-grained named entity recognition in legal documents. In Maribel Acosta, Philippe Cudré-Mauroux, Maria Maleshkova, Tassilo Pellegrini, Harald Sack, and York Sure-Vetter, editors, *Semantic Systems. The Power of AI and Knowledge Graphs*, pages 272–287, Cham, 2019. Springer International Publishing.

[Mas20] Hamza Massaoudi. Crf_ner. https://github.com/hamzamassaoudi/CRF_NER, 2020.

[MM00] Peter Martin and Mike Martin. Design und methodik der interdisziplin"aren l"angsschnittstudie des erwachsenenalters. In *Aspekte der Entwicklung im mittleren und h"oheren Lebensalter*, pages 17–27. Springer, 2000.

[Neu16] Clemens Neudecker. An open corpus for named entity recognition in historic newspapers. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4348–4352, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA).

[NS07] David Nadeau and S. Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30:3–26, 2007.

[NS10] Arvind Narayanan and Vitaly Shmatikov. Myths and fallacies of "personally identifiable information". *Commun. ACM*, 53(6):24–26, June 2010.

[SAVB15] Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90, Parramatta, Australia, December 2015.

[Sch97] R.R. Schaller. Moore's law: past, present and future. *IEEE Spectrum*, 34(6):52–59, 1997.

[Set04] Burr Settles. Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proceedings of the international joint workshop on natural language processing in biomedicine and its applications (NLPBA/BioNLP)*, pages 107–110, 2004.

[SM03] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *CoRR*, cs.CL/0306050, 2003.

[SMS+18] Martin Schiersch, Veselina Mironova, Maximilian Schmitt, Philippe Thomas, Aleksandra Gabryszak, and Leonhard Hennig. A German Corpus for Fine-Grained Named Entity Recognition and Relation Extraction of Traffic and Industry Events. In Nicoletta Calzolari (Conference chair), Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 7-12, 2018 2018. European Language Resources Association (ELRA).

[Sun95] Beth M Sundheim. Overview of results of the muc-6 evaluation. 1995.

[TKS02] Erik F. Tjong Kim Sang. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*, 2002.

A. Appendix: The table of researched datasets

The appendix serves its purpose to give a more holistic overview of the researched datasets in the thesis. Due to the space limit, for more detailed information, such as length of datasets, please refer to the full table under path `data/README.md`. The datasets used in the thesis are introduced in detail in 2.1.

Dataset	Domain	Language	NER-tags	PII-relevant tags	Included	Notes
CoNLL2003	News	Eng & Ger	enamax + MISC	PER, LOC	Yes	English dev not found
Conll2002	News	nl & es	enamax + MISC	PER, LOC	In NLTK	
WNUT17	Social media	English	See 2.1.3	TBD	Yes	
sec-filings	Finance	English	enamax + MISC	PER, LOC	Yes	
GermEval 2014	Wiki+News	German	enamax + MISC	PER, LOC	Yes	
Europeana News	News	Various	enamax	PER, LOC	Yes	
LegalER	Legal	German	enamax + MISC	PER, LOC	Yes	Fine-grained tagging system
NEMGP	Politics	German	?	?	Yes	OpenNLP format. Preprocessing required
WikiNER	Wikipedia	Various	?	?	Yes	Format different. Not in UTF-8.
DFKI SmartData	Various	German	?	?	Yes	.json format.
DBpedia abstract	?	German	?	?	No	Very complicated format.
Benikova et al. 2014	?	German	?	?	No	Included in GermEval 2014.
Tübingen Treebank	?	German	?	?	No	Link does not work, not found
EUROPARL transcripts	?	German	?	?	No	Link does not work, only raw data found
DAWT dataset	?	German	?	?	No	Link and GitHub repo do not work

Table A.1: An overview of the researched datasets of this work