WU WIRTSCHAFTS UNIVERSITÄT WIEN VIENNA UNIVERSITY OF ECONOMICS AND BUSINESS

EQUIS ACCREDITED

Master Thesis

Chose your dataset wisely! The effects of dataset selection on the evaluated performance of named entity recognition tools

Rositsa Ivanova

Subject Area: Information Systems

Studienkennzahl: J 066 925

Supervisor: Johann Mitlöhner

Co-supervisors: Sabrina Kirrane, Marieke van Erp

Date of Submission: 30.05.2021

Department of Information Systems and Operations, Vienna University of Economics and Business, Welthandelsplatz 1, 1020 Vienna, Austria

DEPARTMENT FÜR INFORMATIONSVERARBEITUNG UND PROZESSMANAGEMENT DEPARTMENT OF INFORMATION SYSTEMS AND OPERATIONS

Contents

List of Figures

List of Tables

Abstract

In recent years, automated named entity recognition (NER) in the domain of English literature has been explored through the creation of domain-specific tools such as BookNLP. For the performance evaluation of such tools, researchers have created domain-specific annotated benchmarking datasets (i.e. gold standards). However, even within the same domain the datasets can address different purposes such as the construction of conversational networks, and coreference resolution for the extraction of social networks. This has led to the creation of gold standards within the domain of English literature, which follow different annotation guidelines and thus do not have a unique definition of the individual named entity types (e.g. person). In this thesis we take a closer look at existing gold standards in the domain of English literature. To better understand the differences between the datasets, we select two existing annotated datasets, which have the same purpose (i.e. coreference resolution), yet follow different annotation guidelines. Further, we create two additional gold standards, one of which follows annotation guidelines created for the domain of English literature, the other being one of the most frequently used annotation guidelines in NER across domains (i.e. CoNLL-2003). We evaluate the performance of two NER tools, one domain-specific and one general-purpose NER tool, using the four gold standards, and analyse the sources for the differences in the measured performance. Lastly, we discuss challenges and opportunities, which we have recognised throughout the annotation and evaluation process.

Acknowledgements

I thank my family, who are stubbornly trying to understand what I am studying, and my friends for their continuous support and great patience. Most of all, thanks to Matko, who survived the writing of my first bigger work and never stopped encouraging me. Thanks to my colleagues and friends, with whom I have had the pleasure of sharing coffee breaks, corridor talks and Prater walks.

The writing of this thesis would not have been complete without the continuous feedback and shared knowledge of my supervisors, which I truly appreciate. Big thanks also goes to Martin Trajkov and Martin Beno for their incredible help with the annotation of the novels. I am very grateful to David Bamman, Ikuya Yamada, and Alan Akbik for sharing their insight regarding their research.

Thanks to You - the person who is looking for their name in the acknowledgements section. Finally, I would like to thank the great minds, who create and supply us with amazing ice cream, which brings us even through the most frustrating moments.

Chapter 1

Introduction

Increased digitalization and the introduction of computers to assist with regular day-to-day activities have led to the generation of large corpora of unstructured text. Currently, the majority of new articles in journals, newspapers, magazines, and blogs are published digitally. In addition, old historical documents, books, articles, etc. are scanned and digitalized via optical character recognition (OCR). This ever-growing data collection has led researchers to create and continuously improve natural language processing (NLP) tools, which allows one to draw knowledge from such unstructured texts.

One subtask of NLP is named entity recognition (NER), which focuses on recognizing entities such as people's names and geographic locations. On a day-to-day basis NLP, and in particular NER, is used in various domains. Spam filters, for example, analyse incoming mail to detect suspicious patterns of unwanted messages. Companies process unstructured text to enrich their knowledge bases [59]. Home assistants (e.g. Alexa[1], Siri[2]) do not use predefined phrases anymore, instead they parse the speech and retrieve the relevant information to answer questions or perform desired actions [29]. Furthermore, devices that make the life of impaired people easier, heavily rely on the improvement of NLP technologies, which they make use of. Visually impaired people, for example, often make use of digital assistants, who need to be able to simultaneously understand what the person needs and provide them with accurate help [11]. One further domain, which we focus on in this thesis is the analysis of English literary texts.

Over the years, researchers and engineers have chosen various approaches to improve the recognition and linking of named entities. These different approaches include using deep learning [36], providing the tools with more

[1] `https://www.alexa.com`

[2] `https://www.apple.com/siri/`

Note: All links in this thesis have been last accessed on 28.05.21

annotated training data, making use of technologies such as linked databases to link entities despite diverse name variations [75], enriching the used lists of aliases of entities (i.e. gazetteers) and entity mappings (i.e. dictionaries) for better linking between entities [30]. Most of those approaches have indeed led to a significant improvement in the performance of NER tools. However, as we showcase in the following examples from one NER domain, there is still room for improvement in many aspects.

The majority of NER tools struggle to perform well when the entities in the texts contain specific characteristics. In the domain of novels in the English language, for example, Dekker et al. [15] observed a poor performance in off-the-shelf tools when names contained characters, which are used in an unusual manner for the particular language (e.g. d'Artagnan). In historical letters, Kim and Cassidy [40] described the lack of cues to differentiate between names and ordinary words as one the bigger challenges of the analysis. Furthermore, in their work on NER in old English novels, Woldenga-Racine [72] concluded that the most frequent cause for incorrect NER are capitalized words, which are non-named entities. Many tools rely on gazetteers and rules for NER. However, gazetteers and the machine learning methods used to enrich them are mostly based on modern English, making an analysis of old documents, including books difficult and less accurate. In addition, exceptions to defined rules require the adding of new rules by domain experts [31].

1.1 Research Questions

To address the aforementioned challenges we take a closer look at existing annotated NER datasets in the domain of English literature. In particular, we first take a look at the existing datasets for the literary domain. Next, we compare the measured performance of NER tools using such annotated datasets as a means to detect the differences between the datasets. In this step, the thesis takes a closer look at the effect of different annotated datasets on the measured performance of the tools. Lastly, we discuss the characteristics of annotated datasets, which may lead to such differences between them. The main research question, which we address in this thesis is the following:

How reliable is the use of the existing gold standard datasets for the evaluation of off-the-shelf Named Entity Recognition tools in the English literature domain?

We approach this overarching research question by addressing the following three more specific sub-questions:

1. Which annotated datasets are suitable for the evaluation of off-the-shelf tools for Named Entity Recognition in English literature?

2. How does the use of different gold standard datasets created for the domain of English literature affect the measured performance of Named Entity Recognition tools?

3. What characteristics of a gold standard dataset should be considered when evaluating the performance of Named Entity Recognition tools?

1.2 Design Science Methodology

In this thesis, we observe the creation of new annotated datasets for the evaluation and improvement of NER tools as a continuous problem solving process. This view resembles that of design science research. *"The fundamental principle of design-science research (...) is that knowledge and understanding of a design problem and its solution are acquired in the building and application of an artefact"* [27].

Design Science aims at offering researchers a better understanding of problems and thus helping them to better address existing issues [25]. The goal of this thesis is to better understand the existing gold standards in the English literary domain by evaluating NER tools using those different annotated datasets. For this purpose, we adopt the steps of Design Science Research suggested by Peffers et al. [50]. Figure 1.1 has been adapted from the Design science research process (DSRP) model proposed by Peffers et al. [50] to correspond to the research question targeted by our work.

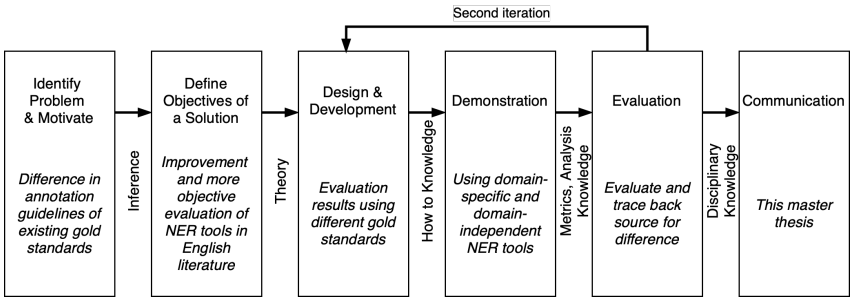

Figure 1.1: Design Science Research

First, we identify the problem. NER in the domain of English literature has not yet been explored widely. However, there already exist tools such as

BookNLP [8] and LitNER [10], which address the domain-specific needs of NER in English literature, as previously recognised by Cranenburgh et al. [64] and Rösiger et al. [54]. Due to the need for training and evaluation data for the domain, researchers have created annotated datasets [6, 15, 63]. Such datasets are typically used for the training and evaluation of tools. We identified certain differences in the annotated datasets (e.g. different named entity types). By selecting and addressing this problem, we contribute to the domain through a detailed analysis of the differences and their sources. An improved solution for the creation of annotated datasets in terms of quality, standardisation of named entity type definitions, etc. for the domain could enable a better (i.e. more objective) evaluation of the performance of domain-specific tools (e.g. BookNLP [8] and LitNER [10]) and could be beneficial to the training of such tools. We base our analysis on the results we derive from an evaluation of two tools (i.e. BookNLP [8] and Flair [5]) using two such annotated datasets (i.e. LitBank [6] and OWTO [15]). Following the analysed discrepancies between the datasets, we discuss the existing alternative approaches for the creation of such annotated datasets (e.g. using different annotation guidelines).

In a second iteration of the Design and Development, Demonstration, and Evaluation steps, we create two new annotated datasets following two separate annotation guidelines. By using them for the evaluation of the two selected tools and increasing the variety of characteristics in the annotated datasets, we gain better insight in the differences between the measured results for the tools, when evaluated using various annotated datasets. Lastly, we analyse the sources of the differences and discuss potential steps that could be taken as a means for creating more unified, and thus more reliable annotated datasets for the English literary domain.

1.3 Structure of the Thesis

In this thesis we contribute to NER in the domain of English literature by taking a closer look at existing annotated datasets. First, Chapter 2 introduces the theoretical background of the thesis. In Chapter 3 we discuss the state of the art of annotated corpora and NER tools, used in the literary domain. Then, we focus on the annotation guidelines, which are used for the creation of the existing datasets. We expand the collection of annotated datasets by creating two new annotated datasets. Following, in Chapter 4 we evaluate the performance of one domain-specific tool and one general-purpose tool when tagging English novels. For this purpose we use off-the-shelf tools - these are tools, which are ready to use (e.g. models are trained) to carry out

NLP tasks [72]. Lastly, in Chapter 5 we analyse any differences detected in the evaluation results and discuss the effect of various dataset characteristics such as the use of annotation guidelines on the evaluated performance of tools.

Chapter 2

Background

Understanding the meaning of written or spoken sentences is a task that for most humans may seem trivial. As long as we understand the words and know the context, it appears easy to comprehend the intended meaning of natural language. Yet, in our everyday lives we might frequently stumble upon situations, in which a person's spoken statement might get misunderstood. At times, this may happen, because we are not familiar with the typical way a certain person expresses themselves or because we heard a statement completely out of context. Other times, we might find ourselves browsing the web and reading the comment of a stranger in a sarcastic tone, only to later realize that the person meant to express themselves in an unsarcastic manner.

2.1 Natural Language Processing

Text does not only confuse people in their understanding of specific statements; when we use machines to process natural language their performance might also be influenced by the intention, context, and goal of a sentence [37]. Computer algorithms are often created for specific purposes or trained on specific corpora of text, which makes them understand specific styles of natural languages better than others [71].

2.1.1 Theoretical Foundation of Natural Language Processing

Due to the omnipresence of text, the research domain of NLP has been of interest to many research fields such as marketing, history, and social sciences. Its main task is to automatically analyse and synthesize human language as

opposed to artificial languages (e.g. programming languages) [31, 43]. Despite the fact that some of the first steps in the direction of understanding human language (i.e. natural language) have been made decades ago, many problems are yet to be solved [24]. To introduce the field of NLP, we give a brief definition of some core linguistic concepts.

Syntax and semantics. Syntax refers to the grammatical rules of a language, while semantics studies the meaning of sentences. When it comes to formal notations (e.g. mathematics, logic) the semantics and the syntax of statements are held together. Based on the structure of a statement, its semantics can be clearly understood [31]. This is not the case when it comes to natural languages. Instead, in natural language two sentences with identical syntax structures, but different words, might have different meanings. For example, the sentences: "Time flies like an arrow" and "Fruit flies like a banana" are syntactically the same. However, the choice of different words in two syntactically equal sentences could change the meaning of the entire sentence.

Pragmatics and context. The pragmatics of a language does not necessarily target the literal meaning of words, but rather observe what the intended meaning of statements is. This is very much context-dependent, as different contexts (e.g. different domains) might entirely change the intended meaning of a statement [37]. If we consider the example "She walked by the shelter with two dogs", there are several ways to understand the sentence. Based on the context, the two dogs might belong to the subject in the sentence (i.e. she) or there might be two dogs in the shelter.

2.1.2 Steps of Natural Language Processing

When processing natural language we typically break longer texts into paragraphs, sentences, and finally, words. In order to be able to do that, certain steps need to be taken, which are often more complex than what one may intuitively expect. Therefore, the tasks are typically segmented into a pipeline. Based on the targeted task and the domain of the text, different tools often adapt the choice and order of the exact steps of the process. The following steps are the most commonly used in NLP. The structure of this section has been taken from the article "Natural language processing is fun!" by Geitgey [23].

Sentence Segmentation. The first task is to split the text into separate sentences. While we may decide to do so based on the fact that many English sentences end in a dot, the task becomes difficult, when we take into consideration that a dot might also have other uses (e.g. abbreviations) [47]. Additionally, we expect a sentence to begin with a capital letter, however not every capital letter is the first word in a sentence [31]. Even in the case when all of those aspects are taken care of, it may not always be possible to analyse each sentence on its own, taking it out of context, as references could be lost once consecutive sentences are analysed in isolation.

Word Tokenization. Next is the step called "tokenization", in which we want to split the sentences into meaningful units of characters (i.e. tokens). Individual words in English are split following the rule that they are separated by an empty space. In addition, one may consider punctuation as a separator of words. Yet tokens do not always represent an instance separated by others via white space or punctuation [47]. They might consist of logically connected characters such as in "home office" or "multi-token". Often this step is completed with the help of defined rules, finite state machines, statistical methods, etc [31].

Predicting Parts of Speech for Each Token. With little practice most humans can learn to recognize what parts of the speech (POS) (e.g. noun, verb) individual words belong to. However, teaching machines how to recognize those, is a task that can be solved by taking various approaches. Different contexts might lead to different meanings of the same sentence [43]. In the example of "The first time he was shot in the hand as he chased the robbers outside" [42], it might be beneficial to use multiple tags instead of a single one when defining the words "first time shot in the hand". Two possible tags for those words would be "JJ NN NN IN NN" and "RB VB VBD RB VB"[1]. The two main approaches to parts of speech prediction are rule-based and stochastic [31].

Text Lemmatization. Words may use different inflections, based on the sentences they are placed in (e.g. university vs. universities, has vs. have). In order to recognize that they refer to the same item or that they have the same meaning, we find the root of the word commonly referred to as a lemma. Lemmatization is typically done by mapping the individual words to tables consisting of lemmas and their use in different words [71].

[1]JJ - adjective, NN - noun, IN- preposition, RB - adverb, VB - verb, VBD - verb past tense

Identifying Stop Words. Some words in sentences (such as "the", "and") are not considered to be as important as others. They are called stop words and are typically the most frequently appearing words in texts. The stop words are usually standard, in the sense that they are pre-defined in lists of words provided by tools (e.g. Natural Language Toolkit [39]) and might need adaptation based on the texts they are applied to. For example, when looking for music bands in text, it might make sense to consider the appearance of names such as "The Who" or even "The The" [31].

Dependency Parsing. In the next step we parse the dependencies in the sentence. This means that we try to find out how the words are connected. Parsing relies on grammatical rules, which are used to define phrase and sentence structures. In contrast, the semantic analysis looks at the roles that phrases and words play in a sentence. The output of dependency parsing is a tree, starting from the main verb, which represents the root, and identifying the parent word of each word. In addition, words are assigned a role in the sentence (e.g. subject, attribute) [43].

2.2 Named Entity Recognition

Named Entity Recognition refers to the identification of proper names in unstructured texts. The term became widespread since the Sixth Message Understanding Conference (MUC-6), where a task of Information Extraction was presented, *"which basically involves identifying the names of all the people, organizations, and geographic locations in a text"* [26]. The final challenge of the conference further involved the detection of time, currency, and percentage expressions. In addition to the task of recognizing the entity text spans, the entities had to be tagged in two categories called ENAMEX and NUMEX. ENAMEX represents the groups from the initial task, including people, organizations, and geographic locations and NUMEX represents the extended group, including time, currency, and percentage.

Nowadays, the two tasks of named entity recognition and classification (NERC) have slowly merged into the umbrella term of named entity recognition. This subtask of NLP has been ported to and applied in various natural languages and domains [18, 32, 66], and the scope of entity categories has been increased to cover more entity types from different domains [58].

The task of named entity linking (NEL) is often discussed alongside NER, yet the two still represent two separate tasks. The challenge tackled by NEL is that it builds upon the differentiation between classes of entities, which NER produces, and further aims to differentiate within the individual classes [43].

An example for such task is the linking of different names and references to the same person (e.g. "The Prime Minister" and "John Smith"). Further, unstructured texts are full of pronouns (e.g. "he", "she") and co-referent expressions (e.g. "their", "the 35 year old"). In contrast to NEL, which aims to link entities between documents and might even use external resources (e.g. DBpedia, Wikidata), co-reference resolution aims to link entities within one and the same document (e.g. "he" and "John Smith").

2.2.1 Entity Types

Over the years, various standards for the annotation of texts and the evaluation of NER tools have been created. The need for those originates from differences between natural languages (e.g. Chinese, English, Hungarian), but also between individual domains (e.g. literature, biology, conversational transcription). For the domain of chemistry for example, Corbett et al. [13] propose a set of annotation guidelines, which include none of the entities defined by MUC-6 [26]. Instead, they define five entity types (i.e. chemical compound, chemical reaction, chemical adjective, enzyme, chemical prefix) that are most important and beneficial for an extraction from chemistry publications. In a similar manner, archaeologists make use of the entity types time and location, but replace the rest of the ENAMEX und NUMEX types with artefact, context, material, and species [9]. Apart from the dissimilarity between domains, the various natural languages require individual guidelines due to their different syntax, morphology, etc. This means that the annotation guidelines used for English would differ from those for Chinese or Catalan for example [65]. It is important to follow annotation guidelines as those help in the creation of unified and comparable datasets. Furthermore, they could be used as an indication of whether or not certain annotated datasets are fitting for a particular domain (i.e. if the definitions of the entity types are suitable).

Currently, the most frequently used annotated datasets are the dataset of the Conference on Computational Natural Language Learning (CoNLL) task in 2003 [61] and OntoNotes [41]. CoNLL-2003 was based on the annotation guidelines of MUC-7 [12, 56]. Table 2.1 depicts the entity types and presents their definitions as per the standards. While eight of the entities are covered by both guidelines, their definitions often differ. For types of locations, for example, OntoNotes uses two entity types - gpe and location, while MUC-7 covers both under the type location.

In this work we focus on the entity person, the definition of which does not differ much in a comparison between these two guidelines. However, while it is relatively clear what falls under the definition of a person in a

Table 2.1: Entity Types in MUC-7 and OntoNotes v5

Entity Type	**MUC-7**[a]	**OntoNotes v5**[b]
PERSON	named person, family, or cetain designated non-human individuals	people, including fictional
ORGANIZATION	named corporate, governmental, or other organizational entity	companies, agencies, institutions, etc.
LOCATION	name of politically or geographically defined location (cities, provinces, countries, international regions, bodies of water, mountains, etc.) and astronomical locations	non-GPE locations, mountain ranges, bodies of water
NORP	NA	nationalities or religious or political groups
FACILITY	NA	buildings, airports, highways, bridges, etc.
GPE	NA	countries, cities, states
PRODUCT	NA	vehicles, weapons, foods, etc. (Not services)
EVENT	NA	named hurricanes, battles, wars, sports events, etc.
WORK OF ART	NA	named documents made into laws
LANGUAGE	NA	any named language
DATE	complete or partial date expression	absolute or relative dates or periods
TIME	complete or partial expression of time of day	times smaller than a day
DURATION	a measurement of time elapsed or period of time during which something lasts	NA
MONEY	monetary expression	monetary values, including units
MEASURE	standard numeric measurement phrases such as age, area, distance, energy, speed, temperature, volume, and weight, plus syntactically-defined measurement phrases	NA
PERCENT	percentage (a fraction expression in terms of hundredths	percentage (including "%")
CARDINAL	a numerical count or quantity of some objects (in form of whole numbers, decimals, or fractions)	numerals that do not fall under another type
QUANTITY	NA	measurement, as of weight or distance
ORDINAL	NA	"first", "second"

[a] Definitions taken from [28]
[b] Definitions taken from [2]

newspaper article, this is not always the case in literary texts. In the context of novels, we typically speak of characters, yet based on the definition of it a character could be a person (e.g. Alice), an animal (e.g. The White Rabbit), or even a personified object (e.g. a talking car). When it comes to the more specific definition of what a character in a novel is, a consensus seems to be needed. As summarised by Bamman et al. [8], different views on a character (e.g. referential, formalist) have different and often conflicting definitions of the term. To the best of our knowledge, there are no guidelines for NER in the literary domain, which define an entity type character.

2.2.2 Challenges

Since MUC-6 took place in 1996 [26], a wide variety of tools for different purposes have been created and improved [24, 36, 71, 73]. However, the recent literature still identifies certain common shortcomings of NER tools. Due to the fact that in research NER and NEL are sometimes merged into an umbrella term and observed together, some of the challenges are also interconnected.

In the following list, we narrow down the scope of challenges that hinder or reduce the performance of NER tools to detecting the person entity type in the literary domain and in those domains, which are as closely connected as possible to the scope of novels in the English language.

- One group of commonly occurring issues relates to nontraditional spelling types:

  - One character can go by many names (e.g."The Director", "Mr.", "Mr.Foster") [15];

  - Characters may also have nicknames (e.g. "The Director") [35];

  - Abbreviations are difficult to recognize correctly (e.g. "WM" for "William") [72];

  - Names preceded by a title (e.g. "Mr.", "Dr.") are not always tagged correctly [72];

  - Only part of a name may be tagged (e.g. *"first and last name were present in the text but only one of the names was tagged"* [72]).

- Another group relates to names that do not follow the same "rules" as names in the real world:

  - Authors may make use of the names they give to characters *"in order to convey certain meaning or function"* [35] (e.g. "Mercy");

- Unidentified names that are so-called word names can be used in personification (e.g. “Clock”) [15];

- Literature in general and fantasy writers in particular frequently make up numerous new names (e.g. “Quill Steward”) [72];

- While in Modern English most names begin with a capital letter, foreign names (e.g. “de Bernezan”) and names of fictional characters (e.g. “robot”) may not always be spelled beginning with a capital letter [72];

- People with names not following the spelling conventions of the targeted language can prove difficult to retrieve (e.g. “d’Artagnan”) [15];

- Uncapitalized named entities are also frequently not tagged (e.g. “the cat”) [72];

- On the contrary, capitalized non-named entity words are often recognized as named entity [72].

- Some references to characters may not mention the character’s name. Instead they may use co-referent expressions (e.g. “The man in the store”) or pronouns (e.g. “he”) [15].

- Plural pronoun resolution can also be difficult to link to the correct character (e.g. “they”).

- Some characters may be relatives and therefore share the same last name (e.g. “Ron Weasley” and “Ginny Weasley”) [35].

- The lack of a set definition of a “character” may lead to different results [8].

- Most approaches use external resources such as encyclopedias, gazetteers, and dictionaries in order to recognise names [15, 24, 72]. This may reduce the quality of named entity recognition for novels, which have not previously been analysed.

- First person novels perform significantly worse than their third person counterparts [15].

- The same name can refer to a variety of named entities or events to different classes of entities (e.g. “Europe”, “Paris”) [3, 24].

- The domain and period of English plays a big role in the performance of tools [72].

- Named entities in short sentences, which provide too little context to the algorithm, often remain unrecognized [72].

2.3 Tools

Tools in NLP and NER are frequently separated in two main categories based on the approach taken for their creation. *"Good Old-Fashioned AI"* primarily uses grammatical and syntactical rules, which means that patterns are matched based on known rules. Typically, when complexities in text comprehension are reached, the issue is resolved by adding rules that specifically address them [31, 47]. *"Empirical NLP"*, on the other side, analyses the text using statistical tools, meaning that the foundation of the approach is derived from patterns and associations found in the large corpora of text. In contrast to the rule-based approach, complexities in text comprehension are addressed by using statistical methods, which stochastically decide based on the probability of the options [31, 47].

In terms of the languages processed, NER tools can be language specific and language-independent. Initially most tools were developed for a specific language. Over time, after the announcement of the CoNLL task in 2003 [61] and even more so with the introduction of BERT [16], language-independent NER tools began drawing attention. Due to the fact that the two types of tools approach the NER task in different ways, they do not always face the same challenges.

Furthermore, some NER tools create a knowledge base as part of their processing pipeline. This knowledge base consists of data collected from gazetteers, dictionaries and similar, which can later be mapped to the targeted text. Some solutions using a knowledge base may expand it with the help of public knowledge graphs such as DBpedia [30]. While enriching the "knowledge" of tools, such approaches may also limit them to the specific languages and modeling choices of the selected knowledge base.

Overall tools in NER can be classified in three main groups: rule-based, learning-based, and hybrid [24]. A rule-based approach follows *"syntactic-lexical patterns to identify and classify named entities"* [24]. Typically, tools applying the technique are domain-specific and require the knowledge of a domain expert. This makes them expensive to create and difficult to reuse or port to other domains. They are also the most likely type of tools to make use of lists of entities (i.e. gazetteers), dictionaries, which map strings to entities, and other sources. Learning-based approaches aim to conquer those shortcomings of rule-based tools by using machine learning techniques. Furthermore, they are separated into supervised, semi-supervised, and unsu-

pervised learning approaches. The main difference between the subcategories lies in the relative amount of labeled training data involved in the creation of the tool, starting from an explicitly labeled corpus for supervised learning to using unlabeled data for unsupervised learning approaches. Lastly, the first two main groups of tools can be merged into hybrid approaches, which aim to make use of the benefits of both groups.

Alongside the many annotation guidelines, which define the named entity types, there are also various standards that can be used for the evaluation of the performance of the tools. Amongst the most used ones are CoNLL [61], OntoNotes [52], and ACE [1]. For the selection of tools for this work we decided to use one tool created for the domain of English novels and to choose another one of the currently best performing tools, which does not target a specific domain. In order to narrow down the decision for the second tool, we followed specific criteria:

- The source code of the tool should be published and be free to use, in order for us to be able to better understand and apply it.
- The tool should not have any specific requirements, which we might not be able to meet (e.g. requirement of a GPU).
- It should be possible to use the tool off-the-shelf, without too many changes (e.g. rewriting parts of the code, having to set up parameters for machine learning algorithms).

Following the order of the evaluated performance of tools, the first tool that comes with source code and does not have any specific requirements[2] is "CrossWeight + Flair". As CrossWeight [68] is to be used as an extension to any NER algorithm, we further looked into Flair [5], which is a stand-alone NER tool. Flair comes with the source code and an extended documentation for users. The tool is pre-trained and allows the use of raw text as input. Furthermore, it is not only used as a stand alone tool, but also in combination with further extensions. As such it is used in four out of the currently six best performing tools measured using the CoNLL-2003 dataset.

[2]At the time of writing of this thesis, the highest scoring tool for NER benchmarked using the CoNLL 2003 NER task is LUKE, which achieved an F_1 score of 94.3 [74]. While the authors of the tool published its source code, it did not meet our further requirements. First, the tool is written specifically for the CoNLL datasets. This means that in its current state it cannot be applied on any other dataset without modification. Second, the tool requires the use of a Nvidia GPU.

2.3.1 BookNLP

Currently, BookNLP is the one of the few NER tools targeting the domain of English novels [8]. It is a tool created for the detection of characters in the literary domain – in particular novels from the 18th and 19th centuries. The main goal of the authors is to create a model, which *"account(s) for the influence of extra-linguistic information (such as author)"* [8]. As such, it relies on the different styles of writing that authors have, as those affect the way characters are portrayed.

The BookNLP model is trained on data originating from online sources, such as Project Gutenberg[3] and HathiTrust[4], and further scanned and OCRed texts. The pipeline of the tool requires the following external tools: Stanford POS tagger [62], linear-time MaltParser [48] for dependency parsing and Stanford named entity recognizer [20]. For coreference resolution the authors differentiate between a character "mention" and an "entity". First, they define a set of existing characters and then map all indirect mentions (e.g. through proper nouns) to those initial characters. For this purpose a Bayesian approach is used.

The pipeline of BookNLP requires a raw text as an input. After processing it through all steps, it outputs a file with tokens and diagnostics files, and optionally can produce an *"HTML file with character aliases, coref and speaker ID annotated"*. The flag *-f* is used for a *"(slower) syntactic processing of the original text"* [5].

For the purpose of this thesis we focus on the tokens that are classified as representing a person. Many gold standard datasets differentiate between the first and all other tokens, which are part of an entity by using the prefix B (i.e. beginning) or I (i.e. inside) respectively in front of the name of the entity (e.g. B-LOCATION, I-LOCATION). Tokens that do not belong to an entity are labeled with O (i.e. outside). This format of tagging is called IOB. The NER tags provided by the BookNLP "tokens" output file do not differentiate between B-PER and I-PER, instead they detect a PERSON. It is important to note that BookNLP includes a column called "supersense", which follows the notation used by WordNet (i.e. lexnames(5WN)) and does differentiate between "B-noun.person" and "I-noun.person". However, sometimes the values depicted in the "supersense" column do not match those in the "ner" column. This means that in some cases when a "PERSON" is tagged in the "ner" column, the "supersense" column instead may hold the value "O", meaning that no entity type was detected, when in fact it should

[3] `https://www.gutenberg.org/`

[4] `hathitrust.org`

[5] `https://github.com/dbamman/book-nlp`

Table 2.2: BookNLP: Mismatch between Columns "ner" and "supersense"

originalWord	pos	ner	characterId	supersense
OLIVER	NNP	PERSON	O	O
MASTER	NN	O	-1	B-noun.person

be either "B-noun.person" or "I-noun.person". Additionally, we observed the opposite case. Examples of both can be seen in Table 2.2, which displays five out of the 16 columns in the token output file produced for the novel Oliver Twist by Charles Dickens.

2.3.2 Flair

Word embeddings are representations (typically as vectors), which can be used to reflect a word in terms of its semantics, context, etc.. They have been widely applied in the field of NER with great success. Starting from the simple approach of creating one embedding per word [45], the models became more complex and offered further embedding approaches such as deriving different embeddings for the same word based on the context it is used in (i.e. contextual string embeddings) [5].

The Flair framework [4, 55] aims to offer all word embeddings types in an easy to use manner by abstracting from the details of their implementation. Flair is *"an NLP framework designed to facilitate training and distribution of state-of-art sequence labeling, text classification and language models"*.

At the time of writing, the tool supports a selection of 8 word and document embeddings[6]. Additionally, the user can decide to combine any of those in a so called "StackedEmbeddings" class. This gives many different options to choose from for the text analysis approach. Furthermore, one may choose to use embeddings on a document level instead of word level. The differentiation between the two gives a different context to the process.

For the training of the model, Flair facilitates a simple setup for accessing publicly available datasets for NLP. Based on the annotation guidelines, the task and the targeted language(s) one can select from nine corpora. The dataset is then downloaded and automatically split into training, testing and development sections. This enables the tool to be usable and comparable in various domains and based on different standards.

[6]classic word embeddings, hierarchical character features, byte-pair embeddings, character-level LM embeddings (i.e. Flair), pooled version of Flair, word-level LM embeddings (i.e. ELMo), ELMo transformer, and byte-pair masked LM embeddings (i.e. Bert)

While Flair allows the user to tune the pipelines to their needs, it also comes as an off-the-shelf solution. For this purpose Flair *"includes a model zoo of pre-trained sequence labeling, text classification and language models"* [4]. The pre-trained models are distributed in two variants - "default" and "fast". The default variants require a GPU to be run on. This means that it can be better fit to the specific case and thus could potentially yield better results. Yet this approach has higher resource requirements that not every user can cover. The fast variants can be ran with a simpler setup using a CPU, which means that the model is not trained again, but is used directly in its pre-trained form. The selection of trained sequence tagger models spans over 16 models for English, 4 multilingual models, 10 models for German, and 10 models for other languages, covers 11 tasks using 17 different datasets[7]. This once again covers different purposes, but also expands the range of languages that the tool can be applied to. For the purpose of NER in English the currently best performing pre-trained model is "ner-large"[8], which scores an F_1 score of 94.09 with the CoNLL-2003 training dataset.

Due to our experiment limitations we select the "ner-large" model from the fast variant. In our case we read the entire text to be tagged as a segment and split it into a list of sentences. The individual sentences are then passed on for the actual tagging. Lastly, we store the token, the predicted tag and the confidence score for the individual prediction. The tags used by Flair follow the BIOES/IOBES (beginning, inside, outside, end, single) format and accordingly use four different prefixes for the tagging of tokens.

- B indicates the beginning of an entity
- I is used for tokens within an entity
- E marks the end of an entity
- S indicates that an entity consists of a single token
- O means that the token does not belong to any entity type

[7] `https://github.com/flairNLP/flair/blob/master/resources/docs/TUTORIAL_2_TAGGING.md` Commit: daa1c02868ebd908cc605cd8bfa0c84b4e050e28

[8] `https://huggingface.co/flair/ner-english-large`

Chapter 3

Annotated Datasets in the English Literature Domain

For the purpose of NER many texts have been annotated over the years. The gold standard of a raw text is typically a manually annotated and agreed-upon benchmark for what entities a tool should be able to detect within that text. Starting with the creation of the MUC [28] and ACE [17] datasets, which among others focus on news articles, web texts, and broadcast conversation, the covered text types needed expansion, as many domains were not yet addressed by the gold standards (e.g. biology, history). Currently, many tools are trained and evaluated using OntoNotes [41, 70], which includes texts from more domains, however the literary domain is not amongst those covered by the corpora. Furthermore, the selection of languages expanded from limited to one language to language-independent NER. The CoNLL-2003 datasets, for example, targeted such language-independent NER using annotated news articles early on and are still frequently used for benchmarking tools.

3.1 Existing Annotated Datasets

Focusing on the purpose of dialogue-based extraction of characters for the creation of a social network, Elson et al. [19] introduce the Columbia QSA corpus of 60 annotated novels. It focuses on dialogue-based extraction of characters for the creation of a social network. Three annotators analysed conversational interactions and detected the corresponding characters. This work targets the domain of literary texts, however it addresses an analysis of the dialogue flow. Recently, authors analysing the challenges of detecting characters and their aliases (e.g. nicknames) for NER tools in the literary

domain created their own datasets for the purpose of their analysis [15, 63]. Simultaneously, a dataset for English novels called LitBank [7] was created as a first step to address the lack of labeled literary texts. Although there exist large annotated datasets in the literary domain in different languages such as German [34], the number of datasets and their scope in English is limited.

For the purpose of this thesis we select two corpora. First, we use the currently most extensive dataset of annotated English novels, called Litbank[1]. Litbank has been created with the intention to address the gap of insufficient datasets specifically for the domain of English literary texts [6]. Second, we select the annotated dataset[2] created in order to extract character networks by Dekker et al. [15]. The authors evaluate the performance of various tools and analyse their shortcomings. We do not make use of the dataset[3] provided by Vala et al. [63] due to the fact that we found the other two collections to follow a format more suitable for our study.

3.1.1 The LitBank Annotated Dataset

LitBank [6, 7, 57] is a dataset consisting of annotated sections of 100 novels. It follows the ACE 2005 guidelines and therefore contains six categories - people, facilities, locations, geo-political entities, organizations, and vehicles.

According to the description of the annotation process in [6] the annotation was done by three people. However, all but 10 novels were assigned to a single annotator each. The remaining 10 texts were used as a means to calculate the consistency between the annotations of the three people. Simultaneously, another publication about LitBank [57], states that the raw texts were annotated by one person only. In addition, five novels were annotated by a second person and served as control units in order to calculate an inter-annotator F_1 score. Both publications state that the scope of the collection is 210,532 tokens within 100 novels. The possibility of a single person vs. three people annotating all texts and producing the same number of tokens is very low. Furthermore, we inspected the commit history of the GitHub repository[4] cited by both papers and did not find an update in the existing annotations in the timespan between the two publications. Lastly,

[1] `https://github.com/dbamman/litbank`
Commit: a371cd678701fc98371355b328a1a6c4b58508a3

[2] `https://github.com/Niels-Dekker/Out-with-the-Old-and-in-with-the-Novel`
Commit: ad31ce1fa515dceabb8febbaa7aa235f3de47ebd

[3] `https://www.aclweb.org/anthology/attachments/D15-1088.Attachment.zip`

[4] `https://github.com/dbamman/litbank`

we compared a previously created GitHub repository for LitBank[5] to the latest above-mentioned repository and did not detect any changes in the files. Therefore, it remains unclear how many novels in the collection of LitBank were double-annotated.

Lastly, LitBank uses multiple layers of annotations. This means that tokens can be a part of multiple entities simultaneously. The authors argue that a flat structure does not cover the needs in annotated literary texts, because for example *"The cook's sister ate lunch contains two PER entities ([The cook] and [The cook's sister])"* [7].

Selection of layer in the LitBank gold standard For the annotation of the raw texts, the authors decided to use the BRAT format[6] and later transform it into a tab-separated values (TSV) format[7]. The TSV files contain multiple layers due to the fact that one token may be (part of) multiple entities. Table 3.1 depicts one such example from the novel "The house of the seven gables", in which the second layer holds one entity consisting of 13 tokens, while the first layer holds two additional shorter entities within those 13 tokens. The number of layers per TSV file (i.e. per novel) is equal to the maximum number of entities that a single token belongs to. For example, if a file has four layers, this means that there is at least one token that belongs to four individual entities.

In addition to the fact that one token may be a part of multiple entity types, it may also be a part of multiple entities of the same type. One such example can be seen in Table 3.2, where certain tokens are part of multiple people entity types (i.e. B-PER and I-PER). Due to the scope of this thesis we reduce the content to the people entity type, which is the most frequently appearing one in literary texts [6]. Those entities are defined as *"a single person indicated by a proper name (Tom Saywer) or common entity (the boy); or set of people, such as her daughters and the Ashburnhams."* [7].

This feature of multi-layered annotations does not have a corresponding flat annotation file representation such as the ones produced by most tools (e.g. BookNLP). In order to be able to evaluate the performance of such tools and to compare this dataset to other annotated ones, we produce a flat version of the file. The flattening of the layers inevitably leads to a loss of information. To address the multiple layers of people entities for the same token we chose to use the people entity with the longest scope. Alternatively, we considered splitting longer entities based on the occurrence of lower-level

[5] `https://github.com/dbamman/NAACL2019-literary-entities`
[6] `http://brat.nlplab.org`
[7] `https://github.com/meizhiju/layered-bilstm-crf`

Table 3.1: Litbank: Multiple Layers containing Different Entity Types

originalWord	**layer1**	**layer2**	**layer3**	**layer4**	**layer5**
the	O	B-FAC	O	O	O
first	O	I-FAC	O	O	O
habitation	O	I-FAC	O	O	O
erected	O	I-FAC	O	O	O
by	O	I-FAC	O	O	O
civilized	B-PER	I-FAC	O	O	O
man	I-PER	I-FAC	O	O	O
precisely	B-LOC	I-FAC	O	O	O
the	I-LOC	I-FAC	O	O	O
same	I-LOC	I-FAC	O	O	O
spot	I-LOC	I-FAC	O	O	O
of	I-LOC	I-FAC	O	O	O
ground	I-LOC	I-FAC	O	O	O

entities within them. For instance, the long entity in the second example of Table 3.2 from the novel "The secret garden" consists of the words *"several of the native servants"*. If we would break it down based on the lower level entities within it, we would create two separate entities "several of" and "the native servants". We did not find this approach fitting, as the first entity "several of" on its own has lost its meaning after being cropped. Therefore, we extracted the longest span of the entities. This approach is also the one chosen for the CoNLL-2003 Shared Task [61].

It is important to note that while a specific approach needs to be chosen for the flattening of a multi-layered gold standard, this inevitably leads to a certain bias in the flattened gold standard. In this regard the user extracting such flat files needs to make a decision, which based on the chosen approach may result in different gold standards. Unfortunately, this defeats the purpose of an "objective truth" to a certain degree. One possible way to reduce this bias would be to include each named entity type at most once per entity. This means that for example "Sofia" in "Sofia's friends" could have the layers B-LOC and I-PER, but not B-LOC, B-PER and I-PER simultaneously. However, this approach would reduce the granularity of the annotated dataset. Alternatively, an additional version, which is flattened could be provided for tools with one-layered output files. In this case, it is essential not to mix the format (i.e. number of layers used) of gold standards, if multiple tools output different number of entity layers.

Table 3.2: Litbank: Multiple Layers containing PER Entities

originalWord	**layer1**	**layer2**	**layer3**	**layer4**
Missie	B-PER	B-PER	O	O
Sahib	O	I-PER	O	O
several	O	B-PER	O	O
of	O	I-PER	O	O
the	B-PER	I-PER	O	O
native	I-PER	I-PER	O	O
servants	I-PER	I-PER	O	O

3.1.2 The OWTO Annotated Dataset

In their work on evaluating the ability of NER tools to extract character entities from English novels in order to build social networks, Dekker et al. [15] created an annotated dataset. Due to the scope of their experiment, they selected 20 modern and 20 old novels and only included the entity type person. They use this differentiation in order to study whether NER tools perform better with modern or old novels. The novels were annotated by two people, both of whom were assigned 20 novels with an average length of 300 sentences. We further refer to this gold standard as the OWTO (Out with the old) annotated dataset.

3.1.3 Comparison of General Characteristics

The following subsection compares the general characteristics of the LitBank and OWTO datasets in terms of dataset size, source of raw texts, annotating approach, purpose, guideline follows, whether or not an initial automated annotation was done, the covered entity types, and the annotation layers. An overview of those can be found in Table 3.3. The size of the datasets varies both in terms of the number of books and the length of the annotated text per book. While LitBank covers more novels, OWTO provides longer sections of annotated text.

Both datasets use Project Gutenberg[8] as a source for the raw texts. In addition, Dekker et al. [15] purchased certain books online. This was a necessary step, as the selection criterion of novels in this case is *"based on Guardian's Top 100 all-time classic novels"* [15], which means that some of them are not available in Project Gutenberg's collection. Both datasets contain annotations created by only one person. In the case of LitBank,

[8] `https://www.gutenberg.org`

according to [7, 57], five novels are annotated by two people and used as control units in order to calculate an inter-annotating F_1 score. The OWTO collection was annotated by two people, yet those were assigned different novels.

Furthermore, LitBank follows *"the guidelines set forth by the ACE 2005 entity tagging task"* [1] and Dekker et al. [15] used BookNLP [8] in order to create an initial annotation as a means to speed up the actual annotation process. Lastly, LitBank offers annotation for the entity types *people, facilities, geo-political entities, locations, vehicles,* and *organizations*, while OWTO focuses on the entity type *person*. In order to allow for the same token in a sentence to be recognised as multiple entity types, LitBank uses multiple layers. This means for example that *"England's queen"* could simultaneously contain a location (i.e. England) and a people (i.e. England's queen) entity type.

The observed differences between the two corpora alone could lead to rather small differences in detecting characters in novels. They both contain the entity type people and span over multiple books. Although the majority of the raw texts originate from Project Gutenberg, the same book may be available in multiple versions. Furthermore, despite the fact that LitBank contains a small number of texts annotated by two people, both collections may include some human error in the annotation (e.g. missed entity).

3.2 Creation of a New Annotated Dataset

In order to better study the differences that the choice of annotation guidelines leads to in terms of tool performance, we create a third version, with which to compare LitBank and Dekker et al. For this purpose we annotated the novel sections which both LitBank and OWTO have annotated. This annotation is not to be treated as a new gold standard, but it serves as a means to better address our research questions and analyse the need for gold standards in the English literature domain.

3.2.1 Extraction of Overlapping Sections

There is a total of 12 novels, which are annotated in both gold standards. Table 3.4 depicts the annotated novels, provided by LitBank and OWTO Within those the overlapping sections of annotated text vary from 1,974 to 2,361 tokens with an average of 2,091 tokens. The texts selected for LitBank often begin with the chapter name, while those by OWTO sometimes skip the chapter name and a few sentences or paragraphs. This leads to the unequal

Table 3.3: Dataset Characteristics for LitBank and OWTO

Dataset characteristics	**LitBank [6, 7, 57]**	**OWTO [15]**
Dataset size	100 novels ca. 2000 words each	40 novels ca. 300 sentences each
Source	Project Gutenberg	Project Gutenberg or purchased online
Annotators	95 novels by one person 5 novels by two people[a]	two annotators 20 novels each
Inter-annotating F_1 score	86.0	NA
Purpose	coreference	coreference resolution, creation of social networks
Guideline followed	ACE 2005[b] [1], OntoNotes[c]	NA
Initial annotation	NA	using BookNLP
Entity types	people, facilities, geo-political entities, locations, vehicles, organizations	people
Annotation layers	multiple	one

[a] Or 90 novels by one person each, remaining 10 by two people each (see 3.1.1).
[b] The annotation process describes certain deviations from the ACE 2005 annotation guidelines (e.g. not including the entity type weapon (WEA)).
[c] The authors also state that they followed the OntoNotes guidelines with certain deviations.

Table 3.4: Overlapping Annotated Novels by LitBank and OWTO

Novel	**Author**	**Nr. of tokens**
Alice's Adventures in Wonderland	Lewis Carroll	2,069
David Copperfield	Charles Dickens	2,033
Dracula	Bram Stoker	2,267
Emma	Jane Austen	2,030
Frankenstein	Mary Shelley	2,364
The Adventures of Huckleberry Finn	Mark Twain	2,170
Moby Dick	Herman Melville	2,204
Oliver Twist	Charles Dickens	1,958
Pride and Prejudice	Jane Austin	2,005
The Call of the Wild	Jack London	2,017
Ulysses	James Joyce	2,006
Vanity Fair	William Thackeray	1,999

length of the 12 overlapping texts.

Due to the high number of differences between the two datasets (e.g. beginning of section, punctuation, spelling) we manually extracted the overlapping sections of raw texts and annotated files. The steps taken in the process are the following:

1. Extract overlapping sections of the annotated files

   After extracting the sections, we ran a simple token comparison of the two gold standard files and noticed the existence of differences between them. Those consisted of the use of different punctuation marks, spelling, and the presence of encoding issues.

2. Correct encoding errors found in the OWTO annotated files with the corresponding entities

   The authors mention that the downloaded raw test files contain certain encoding issues (e.g. dâ€™Artagnan). We replace the affected

characters by their correct version (e.g. d'Artagnan).

3. Correct differences between tokens in LitBank and OWTO

   A detailed inspection of the overlapping raw texts showed that those used by LitBank and OWTO differ. In the case of Moby Dick, for example, LitBank's text is in American English (e.g. honorable) and OWTO's text in in British English (e.g. honourable). A possible reason could be the availability of multiple text versions per novel in Project Gutenberg. Due to the fact that the overlapping novels only consist of "old" books, which are available for free, we assume that Dekker et al. [15] did not have to purchase those for the creation of OWTO. Otherwise, this could be a further explanation of the differences. An exact list of changes in the files can be found in Appendix A.

4. Extract relevant parts of the raw text from LitBank

   We chose to use the raw texts from LitBank due to the fact that the raw texts we retrieved from OWTO partially included encoding errors. Correcting those would have led to further changes to the initial datasets, which we aim to avoid as much as possible.

3.2.2 Annotation Setup

The annotation of the overlapping sections of the 12 novels was done using Doccano v1.2.2[9]. The Doccano repository offers a docker-compose file for simple installation and provides an easy-to-use user interface (UI). The initial setup automatically creates an admin user, who can create projects, upload the required dataset, add labels and give annotation rights to members. We decided to let two people annotate both texts separately. Therefore, we created two users, one for each annotator, and made it impossible for them to see each other's annotations. The explicit separation between the users is essential, as it allowed us to prevent unintended influence between the two annotators.

In the Doccano setup the administrator user was only available to us, meaning that none of the settings can be changed by the annotators. While uploading the raw texts we noticed that Doccano separates the uploaded text into multiple pages based on line breaks. This means that dialogues were often split to one sentence per page. Although we selected the annotation

[9] `https://github.com/doccano/doccano`

to be enabled in the correct order of the text (i.e. as per the novels), we believe that splitting the text in (sometimes) individual sentences may be distracting to the annotators. Therefore we did some small modifications to the paragraph separations.

In terms of paragraphs, we left the longer paragraphs and merged consecutive short paragraphs (e.g.consisting of only one to three sentences). We did this in order to allow for a better annotation experience, as the annotation tool splits the texts into multiple pages based on paragraphs. If we had left the original style of the raw texts unchanged, the annotators would oftentimes need to annotate one sentence at a time instead of a consecutive text[10].

In addition, the text of Alice in Wonderland includes three lines consisting only of stars. We removed those for the purpose of the annotation, as those clearly do not contain any entities. Therefore, they are not available in the output files derived throughout the annotation process.

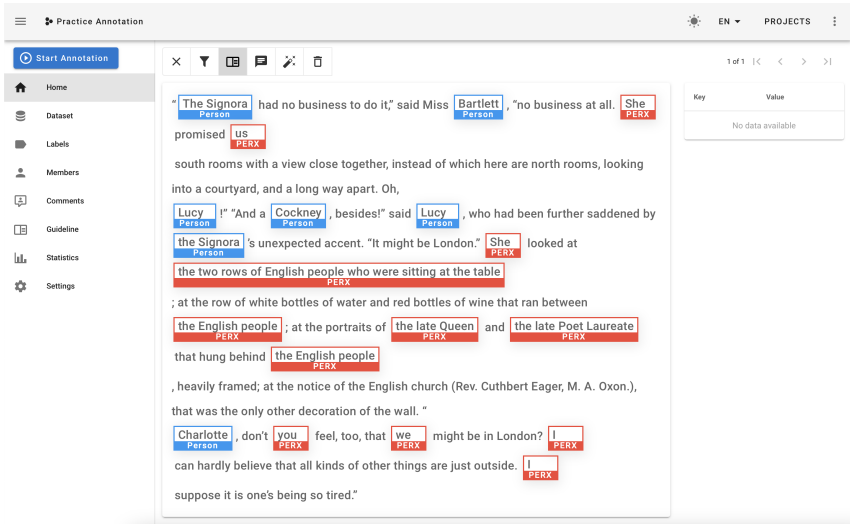

Figure 3.1: Annotation Example using Doccano

Further, Figure 3.2 depicts the steps of the annotation process. First, we gave an introduction to the tool to the annotators and gave them the chance to freely test a practice project, for which we used a text section from another

[10]We did not investigate whether the exact span of the displayed text per webpage during the annotation process has an effect on the decision taken by the annotators.

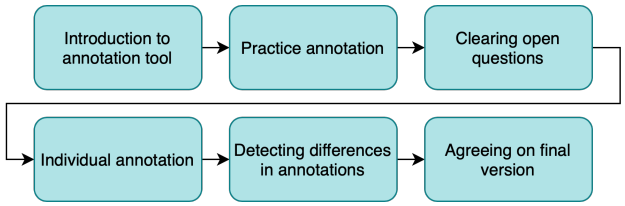

Figure 3.2: Steps of the Annotation Process

one of the 100 raw texts provided by LitBank. An example of the annotation UI can be seen in Figure 3.1. After confirming that the annotators are familiar with the process and that their questions are covered, we proceeded to the actual annotation process. We ensured that the two annotators did not communicate during the individual annotation process. After they annotated all texts, we detected the differences and let the annotators agree on a shared final version.

3.2.3 Annotation Guidelines

The annotators were provided with annotation guidelines to follow throughout the process. They were available on every page with text to be annotated. Overall, we focused on the entity type person. In our guidelines we differentiate between the labels PERSON and PERX. The PERSON label follows annotation guidelines extracted from the MUC-7[11] [12], which the CoNLL-2003 task is based on. We use the guidelines from the CoNLL-2003 task due to the fact that its datasets are amongst the most commonly used ones for the evaluation of tools.

We avoided changing the formulations of the rules and the examples as much as possible, in order to reduce the bias we might introduce to those. This means that the majority of the guidelines are literal extractions of the sections relevant to the person entity type from the original guidelines, which included all entity types. The original guidelines consist of individual rules for the groups of entity types, presenting the taggable and non-taggable instances of entities. In contrast to the original guidelines, we clearly differentiate between, which tokens are to be marked as PERSON (i.e. include) and which are to be ignored, by separating them in two categories. This helped the annotators to clearly identify, whether the entity is to be tagged or ignored. Table 3.5 shows an exemplary subset of the guidelines for the label PERSON.

[11] `https://web.archive.org/web/20060211040221/https://www.nist.gov/speech/tests/ie-er/er_99/doc/ne99_taskdef_v1_4.pdf`

Table 3.5: Subset of Annotation Guidelines for the Label PERSON

Guideline [12]	**Example [12]**
Include	
In a possessive construction, the possessor and (...) substrings should be tagged separately	*John*'s *son*
Acronyms	*JS*, when it stands for John Smith
Ignore	
Aliases that refer to broad industrial sectors, political power centers, etc.	*Uncle Sam*
Laws named after people	the *Gramm-Rudman* amendment

The PERX label extends the PERSON label by accepting more tokens as the person entity type. The PERX label is based on the differences between the CoNLL-2003 guidelines and the annotation guidelines used by Bamman et al. for the creation of the LitBank corpora [6, 7, 57]. We selected those guidelines for the extension of the PERSON label due to the fact that they were chosen with the purpose of creating an annotated dataset for the domain of English literature.

According to the authors of LitBank, their *"annotation style largely follows that of OntoNotes, in defining the boundaries for markable mentions that can be involved in coreference and in defining the criteria for establishing coreference between them"* [6]. With the purpose of coreference, OntoNotes aims to link all mentions of entities in the text to the correct entities. By containing those links, annotated datasets should provide examples that can be used to train computers to automatically extract information through the recognised entities [60]. The main deviations of the LitBank annotation guidelines from those of OntoNotes described by Bamman et al. [6] are the inclusion in LitBank of (i) *"noun phrases that are not involved in coreference"* [6] (i.e. singletons) and (ii) quantified and negated noun phrases. OntoNotes generally does not treat negated noun phrases as taggable, however some exceptions do exist (e.g. "the students" in "none of the students") [60]. Table 3.6 shows an exemplary subset of the guidelines for the label PERX. The complete annotation guidelines can be found in the Annotation Guidelines section of the Appendix.

3.2.4 Annotation Process

The first step of the annotation process was the annotation of a sample text as a practice. This helped the annotators get familiar with the guidelines and the tool. We noticed that it was beneficial to provide the two annotators with

Table 3.6: Subset of Annotation Guidelines for the Label PERX

Guideline[i] **[6]**	**Example**
Include	
Personal pronouns that refer to people	*He* was a noble man
Negated pronouns	*no man*, *none of us*
Ignore	
Bare plurals	*People* need to breathe
Exclamations	*Jesus Christ!*

[i] Additional materials about LitBank were provided to us by David Bamman

a more complicated text with many different types of tokens. This allowed them to familiarise themselves with the process of working with the lengthy annotation guidelines and their structure. At this point, the annotators first stumbled upon a case, in which one token can be tagged both as PERSON and as PERX. Unfortunately, we did not find an option to tag a token with more than one label, therefore the annotators were asked to add one of the entities in such cases as a comment. An example of this can be seen in Figure 3.3. Those entities were then added to the list of regularly tagged entities and can, therefore, not be seen in the Doccano output files within the list of regularly tagged tokens, but as separate comments.

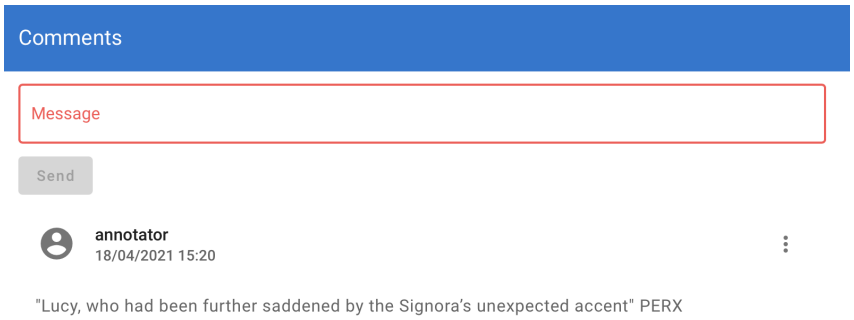

Figure 3.3: Adding Comments using Doccano

Next, the individual annotation done by the two people took place. This step of the process took the longest amount of time, as we avoided pressuring the annotators time-wise. Instead, they could choose their own tempo and approach. The first annotator decided to read the texts slowly and annotate them once, while the second annotator preferred to read the texts quickly

once and then check for mistakes in a second round. Regardless of the approach taken, both annotators reported certain difficulties. It is important to state that, while both annotators understood the importance of the task for NER, neither of them had done annotation previously. Considering their lack of experience in annotating, both annotators found the guidelines for the label PERSON to be more precise and less ambiguous. They reported to have had sometimes experienced issues tagging tokens as PERX. Most frequently, they were uncertain about the beginning and the end of the entire noun phrases describing an entity. Overall, they both stated that more detailed guidelines, including edge cases, which are currently not explicitly targeted by the guidelines, would help them in the tagging process and would decrease their bias.

To evaluate the inter-annotator agreement we used Cohen's kappa [44]. The range of the agreement score is from -1 to +1, where 0 is the agreement, which is to be expected from a random selection (i.e. 50% agreement). Kappa values above 0.60 and below 0.80 are viewed as representing a *moderate* level of agreement. Those between 0.80 and 0.90 are considered *strong*, while values above 0.90 are already viewed as *almost perfect*. In our case, the overall achieved Cohen's kappa scores are within these levels. An overview of the calculated scores[12] is displayed in Table 3.7. The inter-annotator agreement for the PERSON label (i.e. following the CoNLL-2003 guidelines) is in all but one case equal or above 0.90, which puts it in the level of an almost perfect agreement. The range in the Cohen's kappa for the PERX label is from 0.69 to 0.90, meaning that it is distributed in the moderate and strong levels. These results confirm the feedback received by the two annotators, who indicated that the more detailed annotator guidelines for the PERSON label made it easier for them to know which entities are to be tagged.

[12]For the calculation of the results we used the approach provided by `https://github.com/o-P-o/disagree`, which we updated to cover all issues reported in the repository by previous users. The final version of the used code can be find in the repository of this project.

Table 3.7: Inter-annotator Agreement for the New Datasets

Novel	**PERSON label**	**PERX label**
Alice's Adventures in Wonderland	0.88	0.90
David Copperfield	1.00	**0.69**
Dracula	0.90	0.74
Emma	0.97	0.77
Frankenstein	1.00	0.76
The Adventures of Huckleberry Finn	0.97	0.85
Moby Dick	0.95	0.78
Oliver Twist	1.00	0.76
Pride and Prejudice	0.98	0.90
The Call of the Wild	0.92	0.70
Ulysses	0.96	0.79
Vanity Fair	0.90	0.74

Chapter 4

Evaluation of NER Tool Performance

For the performance evaluation of NER tools we make use of two existing annotated datasets from the literary domain (i.e. LitBank and OWTO) and the two new datasets, created for the purpose of this work. We evaluate each of the tools using all four gold standards. Next, we take a closer look at the measured performance of the tools. We analyse the differences in the individual results by taking a closer look at the characteristics of the individual gold standards.

4.1 Evaluation Metrics

For the purpose of evaluating the tools, we use the metrics that over the years have become the standard in the field of NER [26, 58, 73]: those are precision, recall and F_1.

Precision represents the percentage of correctly recognized entities out of all entities tagged by the tool we are evaluating. Here it is important that the recognition is exact. This means that a tool should tag correctly all tokens that belong to an entity. For example, if a person entity consist of the multiple tokens "The Godfather" according to the gold standard, the tool should tag both "The" and "Godfather" in the same manner for the recognition to be counted as exact. A partial recognition, such as only tagging "Godfather" is not accepted as correct.

$$Precision = \frac{Ncorrect}{Ncorrect + Nfalse\_positive} \tag{4.1}$$

Recall depicts the percentage of entities correctly tagged by the tool out

of all existing entities as per the gold standard. The existing entities equal the sum of the true positives (i.e. correct) and and the false negatives.

$$Recall = \frac{Ncorrect}{Nexisting} \tag{4.2}$$

F_1 is the harmonic mean of Precision (P) and Recall (R).

$$F_1 = 2 * \frac{(P * R)}{(P + R)} \tag{4.3}$$

In certain cases we need to consider how to handle a division by zero. This could be the case, if the tool has not managed to correctly detect any entity. This leads to the precision and the recall being zero. This means that calculating the F_1 score would require a division by zero. In order to address this rare, but yet existing issue, we take the following approach by the Data Science Group at UPB, Germany[1]:

- In the case that the true positives, false positives and false negatives are all equal to zero, we mark the F_1 score and the Precision as 1, and the Recall as 0.
- In the case that the true positives equal to zero and the false positives or the false negative are bigger than zero, we mark the F_1 score, the Precision, and the Recall as 0.
- In all other cases we calculate the three metrics following the formulas presented above.

4.2 Setup and Steps

We run our experiment in multiple steps as depicted by Figure 4.1. In the first main stage of the process we retrieve the two existing annotated datasets (i.e. LitBank and OWTO) in a fitting format for our experiment. Further, the 12 overlapping sections of the novels are annotated by the two annotators. Lastly, we selected and set up the tools for the evaluation (i.e. BookNLP and Flair).

[1] `https://github.com/dice-group/gerbil/wiki/Precision-Recall-and-F1-measure` Commit: bc74cb233dcd5a90ce3b1fe6bea5f5bac0462e6e

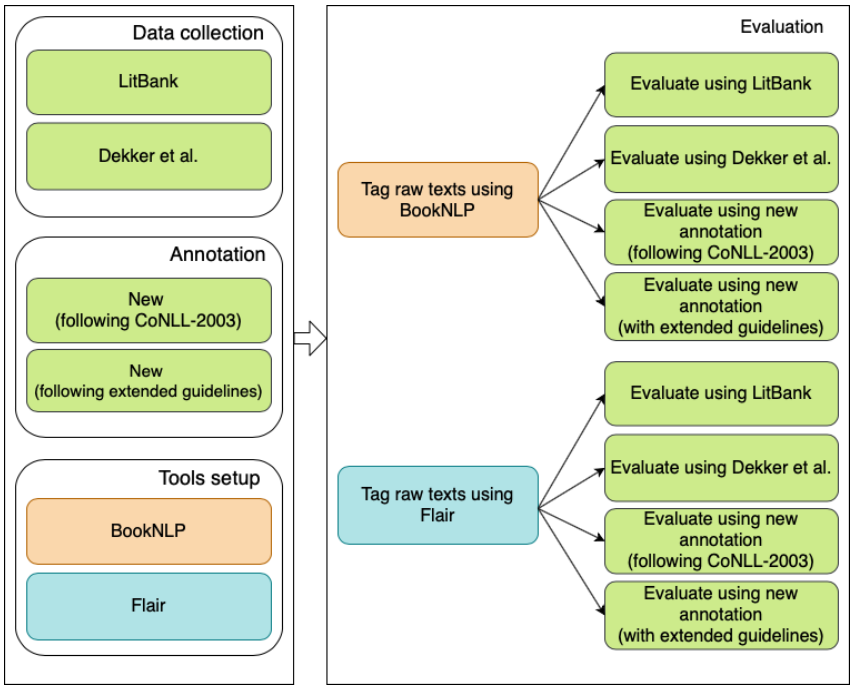

Figure 4.1: Steps of the Experiment

Determining the span of entities. In the second stage, we tag the raw text sections extracted from the LitBank corpora. Those texts are unprocessed (i.e. as taken from the novels). Using the outputs of the tools, we proceed with the evaluation of the performance of the two tools using the four individual annotated datasets. The evaluation is done using the CoNLL-2003 script[2], which offers two main approaches for the evaluation. The token-based approach compares the tags in the gold standard with those in the tool's output token-by-token. The phrase-based approach looks at the phrases recognised by the gold standard and evaluates, whether the tool tagged all tokens within the phrases (i.e. entities) correctly. To select the fitting approach, we have to consider the different tagging formats used by the tools and the annotated datasets.

- BookNLP does not provide any prefixes to the NER tags, which means that every token, contained within an entity of type person is tagged

[2] `https://www.clips.uantwerpen.be/conll2003/ner/bin/conlleval`

Table 4.1: Difference in Entity Span Detection based on a Prefix

original word	**ner (with prefix)**	**ner (without prefix)**
think	O	O
the	B-PER	PER
archangel	I-PER	PER
Gabriel	B-PER	PER
thinks	O	O

as "PERSON".

- Flair differentiates between the prefixes S, B, I, and E, and as such offers an indication about the span of the tagged entities.
- In LitBank the inside, outside, beginning (IOB) format is used, in which the beginning of entities is marked by the prefix "B".
- OWTO uses one label for tokens, which are a part of an entity (i.e. "I-PERSON"). However no further prefixes are used, therefore we treat this format in the same manner as the one in the output files of BookNLP.
- For our new annotation, we distinguish between the beginning of an entity and tokens inside the entity using the IOB format.

The differentiation between those tagging formats is essential as it affects the evaluation approach. For example, when comparing the Flair tags with those of LitBank we could make use of the indications for the beginning of entities in order to clearly identify the complete span of each entity. The explicit use of prefixes reduces the level of bias in the evaluation, as it allows for clear separation of entities even if they are located immediately one after the other.

Table 4.1 depicts the difference between tagging entities using a prefix and omitting it using an example from the novel Moby Dick. Here, the prefix allows us to detect that the annotators recognised two separate entities - "the archangel" and "Gabriel". However, if no prefixes are used, we need to decide how to interpret and use the labels. In this example, the lack of prefixes may lead to treating each tagged token as a separate entity, treating all three tokens as one entity, or splitting the subsequential tokens into multiple tags. We stumble upon this case when working with the tool BookNLP and the annotated dataset OWTO, both of which do not use prefixes for the labels.

In our initial approach of comparing BookNLP, which does not use prefixes, with LitBank, which uses prefixes, we considered treating each token

Table 4.2: Determining the Correctness of Entity Tagging by BookNLP

original word	**gold standard**	**BookNLP tag**
a) exact overlapping		
think	O	O
the	B-PER	PERSON
archangel	I-PER	PERSON
Gabriel	I-PER	PERSON
thinks	O	O
b) one incorrect		
think	O	PERSON
the	B-PER	PERSON
archangel	I-PER	PERSON
Gabriel	I-PER	PERSON
thinks	O	O
c) one incorrect		
think	O	O
the	B-PER	PERSON
archangel	I-PER	PERSON
Gabriel	I-PER	O
thinks	O	O

(i.e. each row in the file) as a separate entity. This approach, however, leads to a rather biased result:

- First, we would give more leverage to longer entities than to shorter ones, due to the fact that we would count each token as a separate observation.
- Second, because we would treat each row as an individual entity, we could not prevent partial correctness to be accepted.
- Further, the LitBank gold standard also treats commas as a part of a PER entity (e.g. a rabbit with either a waistcoat-pocket, or a watch), which also would increase the number of observed entities.
- Lastly, the separation of the hyphen compound words in multiple tokens would contribute to the increase of the number of (in)correct detections.

Therefore, considering amongst others these four reasons, we adapt our approach. In order to be able to evaluate both tools using all gold standards in the same manner, we take the phrase-based approach, which takes a look at an entity as a whole chunk, instead of looking at it token-by-token.

Table 4.2 depicts three exemplary cases of phrase-based evaluation using LitBank as a gold standard and BookNLP as a tool. In case a) the gold standard recognises one entity and the tool has marked all tokens, which belong to those two entities as PERSON. Therefore the tagging by the tool is correct. In case b) the BookNLP's output has one more tag, which is outside of all entities as per the gold standard. As we are looking at the phrase, we evaluate this as one incorrect recognition. In case c) the first two tokens labeled by the tool fully cover the first entity from the gold standard, however the third token is unmatched. Due to the fact that partial correctness is not accepted by the evaluation metrics correctness, the recognition is viewed as incorrect.

Managing inconsistencies in tokens. One inconsistency we observe in the files of LitBank is that some hyphenated words are not split over multiple tokens in the output files. This means that in some cases hyphenated words (e.g. "a-bothering" in The adventures of Huckleberry Finn) are kept together as one token, while in others (e.g. "waistcoat-watch" in Alice's adventures in Wonderland) the words are split into three tokens (i.e. "waistcoat", "-", "watch"). This occurrence becomes an issue in the evaluation step, in which we compare the tool output to the gold standard token-by-token. The reason for this inconsistency in the parsing of hyphenated words might originate from the use of Stanford CoreNLP.

For the creation of Litbank, the authors state that they used the Stanford tokenizer for preprocessing. Considering that the paper describing the annotated dataset creation has been published in 2019, we conclude that the authors have used a version not more recent than 2019.

Stanford CoreNLP v.4.0.0, released on 4th May 2020, introduced *"UD v2.0 tokenization standard for English, French, German, and Spanish. That means "new" LDC tokenization for English (splitting on most hyphens) and not escaping parentheses or turning quotes etc. into ASCII sequences by default."* [3]. This means that due to the use of an older version during the creation of the annotated dataset, hyphen compound words in LitBank may be represented as one token.

At the time of this experiment, BookNLP is using Stanford CoreNLP v.4.1.0. This means that hyphenated words are split into multiple tokens, as depicted in Table 4.3. The use of different Stanford CoreNLP versions for the creation of LitBank and for the tagging of the raw texts using BookNLP leads to mismatches between the tokens in the gold standard and the tool's output. In order to address this inconsistency between the BookNLP output

[3] `https://github.com/stanfordnlp/CoreNLP/releases/tag/v4.0.0`

Table 4.3: BookNLP: Appearance of Hyphens

originalWord	pos	ner
Rabbit	NN	PERSON
-	HYPH	PERSON
Hole	NN	PERSON

and LitBank, we split all hyphen compound words in to separate entities as done by the Stanford CoreNLP v.4.1.0.

The second observed inconsistency originates from encoding issues in the OWTO dataset. The authors describe the encoding issues as originating from the raw text files of the novels [15]. We work around them by replacing the respective tokens such as "dâ€™Artagnan" with their correct version - "d'Artagnan".

Setup. The source code, data and raw results of this experiment can be found at `https://github.com/therosko/Thesis-NER-in-English-novels`[4]. The entire experiment has been executed from a Docker container, the specification of which (i.e. the Dokerfile) have been added to the repository. The container was run on a virtual machine with an Ubuntu 18.04 operating system, and has 8 vCPUs and 16 GB RAM allocated on a QEMU/KVM hypervisor host[5].

4.3 Results

In the second stage of the experiment we evaluate the performance of BookNLP and Flair using the four individual datasets - LitBank, OWTO, the new dataset following the CoNLL-2003 guidelines for the entity type PEOPLE, and the new dataset, which is extended by the annotation guidelines of LitBank targeting the domain of English novels. Table 4.4 and Table 4.5 display the precision, recall and F_1 scores for BookNLP and Flair respectively. First, the scores per novel are presented and then at the bottom of both tables we

[4]A permanent link to the most recent location of the repository can be found at `https://rivanova.org/master_thesis`

[5]The hardware specifications of the host:

- CPU: Intel(R) Xeon(R) Gold 6126 CPU @ 2.60GHz
- RAM: 384 GB DDR4 2666 MT/s
- NVMe PCIe Storage

summarise the scores using the mean, standard deviation, and the median for each evaluation metric and annotated corpora.

Table 4.4: Evaluation of BookNLP

Novel	LitBank			OWTO			New (CoNLL)			New (Ext)		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
Alice in Wonderland	80.00	54.05	64.52	92.00	74.19	82.14	100.00	80.65	**89.29**	100.00	12.89	22.83
David Copperfield	100.00	18.06	30.59	44.44	85.71	**58.54**	0.00	0.00	0.00	0.00	0.00	0.00
Dracula	45.45	10.87	17.54	14.29	33.33	20.00	45.45	35.71	**40.00**	45.45	2.16	4.12
Emma	86.67	38.24	53.06	83.10	98.33	**90.08**	40.00	36.73	38.30	37.78	6.59	11.22
Frankenstein	77.78	9.33	16.67	41.67	100.00	58.82	77.78	70.00	**73.68**	77.78	2.47	4.79
The A. of Huckleberry Finn	82.61	33.33	47.50	73.53	78.12	**75.76**	60.87	50.00	54.90	56.52	4.91	9.03
Moby Dick	71.43	6.58	12.05	37.50	100.00	54.55	71.43	62.50	**66.67**	42.86	1.42	2.75
Oliver Twist	73.33	11.96	20.56	70.00	100.00	82.35	93.33	87.50	**90.32**	86.67	6.81	12.62
Pride and Prejudice	95.74	42.06	58.44	73.08	98.28	**83.82**	31.91	31.25	31.58	29.79	4.58	7.93
The Call of the Wild	84.21	14.81	25.20	94.74	41.86	**58.06**	84.21	37.21	51.61	78.95	6.52	12.05
Ulysses	92.98	50.48	65.43	81.58	98.41	89.21	96.49	94.83	**95.65**	92.98	18.21	30.46
Vanity Fair	70.15	31.33	43.32	74.59	88.35	**80.89**	22.39	18.99	20.55	14.93	4.50	6.92
Mean	80.03	26.76	37.91	65.04	83.05	**69.52**	60.32	50.45	54.38	55.31	5.92	10.39
Standard deviation	14.46	16.89	19.75	24.8	23.1	20.34	32.32	29.02	29.9	32.14	5.11	8.65
Median	81.31	24.7	36.96	73.31	93.32	**78.33**	66.15	43.61	53.26	50.99	4.75	8.48

At first glance it is noticeable that the results for both tools vary heavily based on the annotation dataset used as a gold standard. For BookNLP we observe an F_1 score range from 0.00 for "David Copperfield" using both new annotations up to 95.65 for "Ulysses" using the new annotation following the CoNLL-2003 guidelines. In the case of Flair this range is from 0.00 for "Dracula" using OWTO up to 96.97 for "Pride and Prejudice" using the new annotation following the CoNLL-2003 guidelines. In order to better understand the discrepancy, we compare the annotation guidelines of the individual datasets.

Comparison of annotation guidelines The annotated datasets, which we use for the evaluation of the tools, do not follow the same annotation criteria. There are many reasons for using different guidelines such as the creation purpose of the annotated dataset. Tagging a text for the purpose of extracting a conversational network, for example, may target different entities than if the purpose was extracting the locations that people can be assigned to. Additionally, the used annotation guidelines may be following a certain standard (e.g. CoNLL-2003, ACE 2005) or may be targeting a specific gap in the previously available annotated datasets.

Table 4.5: Evaluation of Flair

Novel	LitBank			OWTO			New (CoNLL)			New (Ext)		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
Alice in Wonderland	76.92	54.05	63.49	92.31	82.76	82.27	100.00	83.87	**91.23**	100.00	13.40	23.64
David Copperfield	87.50	19.44	**31.82**	6.25	7.69	6.90	6.25	6.67	6.45	6.25	0.49	0.90
Dracula	57.14	8.70	15.09	0.00	0.00	0.00	57.14	28.57	**38.10**	57.14	1.72	3.35
Emma	60.00	26.47	36.73	57.78	68.42	62.65	71.11	65.31	**68.09**	68.89	12.02	20.46
Frankenstein	46.15	8.00	13.64	15.38	50.00	23.53	69.23	90.00	**78.26**	61.54	2.83	5.41
The A. of Huckleberry Finn	72.41	36.84	48.84	62.07	81.82	70.59	72.41	75.00	**73.68**	68.97	7.55	13.61
Moby Dick	88.89	10.53	18.82	33.33	100.00	50.00	88.89	100.00	**94.12**	66.67	2.84	5.45
Oliver Twist	6.67	10.87	18.69	66.67	90.91	76.92	86.67	81.25	**83.87**	80.00	6.28	11.65
Pride and Prejudice	29.41	14.02	18.99	21.57	31.43	25.58	94.12	100.00	**96.97**	88.24	14.71	25.21
The Call of the Wild	88.24	27.78	42.25	88.24	75.00	**81.08**	88.24	69.77	77.92	85.29	12.61	21.97
Ulysses	90.48	54.29	67.86	71.43	100.00	83.33	88.89	96.55	**92.56**	87.30	18.90	31.07
Vanity Fair	37.23	23.33	28.69	36.17	49.28	41.72	55.32	65.82	**60.12**	41.49	17.57	24.68
Mean	61.75	24.53	33.74	45.93	61.44	50.38	73.19	71.9	**71.78**	67.65	9.24	15.62
Standard deviation	27.33	16.42	18.6	31.46	34.1	30.37	25.47	28.56	26.5	25.09	6.44	10.16
Median	66.21	21.39	30.26	46.98	71.71	56.33	79.54	78.13	**78.09**	68.93	9.79	17.04

Table 4.6 gives on overview of the known differences in the annotation guidelines used for the four datasets. The number of the presented differences is limited due to the fact that not all guidelines have the same length and that they do not cover the exact same annotation rules. However, those eight examples clearly indicate that the annotated datasets diverge in their understanding of correctness in the tagging.

While both LitBank and OWTO have been created for coreference resolution in the domain of English novels, they do not treat pronouns, bare plurals and honorifics the same way. The two new datasets annotated for the purpose of this thesis also do not fully match any of the other two datasets. The first one strictly follows the annotation guidelines of CoNLL-2003, while the second one serves as an extension of it, adding the guidelines of LitBank, which target the domain of English novels. It is important to note that the extended guidelines of the new dataset do not exactly match with those of LitBank. This can be seen in the example of honorifics, which are tagged in LitBank, but are not tagged in our new extended dataset. The difference originates from the fact that the annotators worked with two labels (i.e. PERSON for CoNLL-2003 and PERX for the extended guidelines), one of which is seen as an extension of the other. It would have been more difficult and potentially less consistent for untrained annotators to follow and apply two completely different guidelines simultaneously. Therefore, we simplify the task by extending the guidelines defining the PERSON label by more

Table 4.6: Annotation Guidelines for LitBank and OWTO

Annotation guidelines	**Example**	**LitBank [6, 7, 57]**	**Dekker et al. [15]**	**New CoNLL**	**New Ext.**
Generic pronouns	"Everyone knows, **you** don't mess with **me**!" [15]	No	No	No	No
Exclamations	"For **Christ**'s sake!" [15]	NA	No	No	No
Generic noun phrases	"Bilbo didn't know what to tell **the wizard**" [15]	NA	No	No	No
Personification	"the way to hear **the Rabbit** say to itself"	Yes[a]	Yes	Yes[b]	Yes
Single pronoun references	"he"	Yes	Yes	No	Yes
Plural pronoun references	"they"	Yes	No	No	Yes
Bare plurals	"pirates sail ships" [57]	Yes	No	No	No
Honorifics	"Mrs Robinson"	Yes[c]	No	No	No

[a] *"(...) include characters who engage in dialogue or have reported internal monologue, regardless of their human status"* [7]
[b] when referred to by a name
[c] unless in isolation

rules, which define the PERX label. A discussion with the annotators after the completion of the annotation process confirmed that they would have found it more difficult to select the correct label, if they had to simultaneously use two different and partially contradicting guidelines. We make use of this case to showcase how new guidelines may come together - in our case due to the inexperience of annotators and the complexity of the task - how those guidelines may provide a different understanding of what entities are correct, and as such how this may lead to entirely different evaluation results.

As defined by the annotation guidelines of LitBank, only *"characters who engage in dialogue or have reported internal monologue, regardless of their human status"* [7] are to be tagged as "person" entities. The method used for the identification can be described as *"based on animacy, determined through dependencies with "sentient" lemmas from a small dictionary (including for example, say and smile), and gender, assigned through pronomial resolution and a dictionary of gender specific honorifics"* [63].

This approach could be useful for certain tasks (e.g. building or analysing a communication network). However, when using this dataset for a different purpose, it may be important to consider, whether or not this is a fitting approach for the case at hand. In the book Moby Dick by Herman Melville, for example, the story revolves around the main protagonist Ishmael, who is chasing the main antagonist - a white whale called Moby Dick. Despite its central role, the whale does not engage in dialogue or monologue. If our use case would focus on the relationships between characters in novels, we would be missing an essential antagonist.

Another example of such a character is Kattrin in "Mother Courage and Her Children" written by Bertolt Brecht. Kattrin cannot speak, yet is a central character to the story. Those examples show that based on the purpose of the analysis certain guidelines might exclude some essential characters. Therefore, when selecting a dataset as a gold standard it is important to consider the definitions given to a character. For example, a NER tool, which considers Mody Dick and Kattrin to be characters, would perform better when evaluated using a gold standard that considers them to be characters, too. Yet, if a gold standard with a different definition would be used to evaluate those tools, the tool tagging Moby Dick and Kattrin as characters would be evaluated as wrong.

4.4 Analysis

Considering the effect that annotation guidelines have on the results of the evaluated performance of tools, we look at the individual metrics from our experiment in detail. For this purpose we discuss the performance of both tools separately.

If we examine at the F_1 score alone, both tools perform best when evaluated using the new annotation following the CoNLL-2003 guidelines and using OWTO, and they perform the worst using the new extended annotation. One of the main reasons for the poor performance of the tools using the new extended dataset is the fact that the extended dataset considers personal pronoun references (e.g. you, her) to be entities. In novels, such as Ulysses, Pride and Prejudice, and Frankenstein personal pronouns make up around 200 entities per novel, which is around 10% of all tokens in the annotated sections. When tools do not tag those pronouns as correct, their recall drops drastically, even if the precision of the tool is otherwise relatively high. In the case of BookNLP the median of the precision using the new extended dataset is 77.5%, while the recall is only 6.93%. In the case of Flair, the gap is even bigger with a precision of 90.54% and a recall of merely 9.65%. Both cases show a very low F_1 score on average.

Considering that the annotation guidelines of LitBank also include personal pronouns as entities, whenever they refer to (in our case) the entity type person, we would expect the results of the evaluation with LitBank to depict the same shortcomings of the tools as the results of the new extended dataset. However, despite the same formulation of the annotation rule, the LitBank gold standard contains only occurrences of personal pronouns in conjunction with other tokens (e.g. my mother) and not as single token entities (e.g. you).

In terms of precision, the biggest difference between the results using LitBank and the new extended dataset comes from the different approach to honorifics. LitBank includes honorifics as a part of the entity. Due to the fact that we follow the CoNLL-2003 as primary guidelines, we exclude honorifics from the list of taggable tokens in the new dataset. The effect of not including them in the new gold standards can be mostly recognised by the precision results of the novels Emma, David Copperfield, Pride and Prejudice, and Vanity Fair in the case of BookNLP, and of the novels Emma, David Copperfield, and Vanity Fair for the case of Flair. Interestingly, the effects differ as both tools handle honorifics differently. BookNLP appears to tag the majority of honorifics, yet Flair mostly tags unabbreviated honorifics (e.g. Miss) and excludes abbreviated ones (e.g. Mr.). Due to the fact that most honorifics in Pride and Prejudice are abbreviated, we see that Flair clearly scored higher in terms of precision compared to BookNLP. The same effect of the annotation rule about honorifics can be seen in the precision values using the new annotated dataset following CoNLL-2003.

In terms of the precision values observed using LitBank as the gold standard, we noticed only one pattern, which leads to a lower precision value in Alice's Adventures in Wonderland. The gold standard does not consider Alice's cat "Dinah" to be a person entity, however the tools tag the cat as an entity. We do not recognise any other patterns clearly leading to imperfect precision values when using LitBank as a gold standard.

When we look at the recall values achieved by the tools, it is surprising that both tools achieve 100% recall for some of the novels when evaluated using the OWTO gold standard. We manually confirmed that the values are correct and found out that based on the OWTO gold standard the three of the novels - Frankenstein, Moby Dick, and Oliver Twist - have only 4, 3 and 11 person entities respectively. This explains why it is realistic to achieve 100% recall on all three of them. Surprisingly, Flair also correctly tagged all entities in Ulysses without scoring any false negatives despite the section of the novel having 45 entities.

Furthermore, Flair achieves 100% recall also when evaluated using the novels Moby Dick, and Pride and Prejudice from the new annotated dataset following CoNLL-2003. The lowest precision score by BookNLP using this gold standard is for David Copperfield, as all entities in the novel contain a honorific followed by a name and those are not treated as parts of an entity following the CoNLL-2003 guidelines. This is further the reason why the precision equals 0.00% also for the evaluation using the new extended gold standard. Those cases present two edge cases, in which we avoid the division by zero by setting the F_1 score to 0.

"Dracula" is the novel, on which Flair scores the lowest when the OWTO

gold standard is used. The outlier score of 0.00 results from the the fact that the tool did not detect any entity entirely correctly. The tool tags "Mina" as a person twice and once as miscellaneous, while the annotator of the gold standard only treats one of the occurrences as a person. Furthermore, Flair does not tag "Count" as a part of Dracula's name, however in the gold standard both "Count" and "Dracula" are labeled as a person entity type. Lastly, "Jonathan Harker" is also considered as a false positive, however we did not find an explanation of this case. We view the tagging of "Mina" and "Jonathan Harker" with the label "O" as annotation mistakes.

Further, BookNLP and Flair score relatively low in terms of recall when evaluated using LitBank as a gold standard vs. when using OWTO or the new dataset following CoNLL-2003. The main reason for this is that the annotation guidelines of LitBank include common phrases such as "a boy" and require entities to include the entire noun phrases such as "the youngest of the two daughters of a most affectionate, indulgent father" (from the novel Emma). Those entities are tagged neither by BookNLP nor by Flair. As the new extended dataset also applies those rules, the recall values achieved by the tools using it as a gold standard are also low.

4.5 Threats to Validity

The presented results are the output of one approach consisting of multiple steps, each of which influences the final results. One potential major influence on the results is the method of detecting the span of entities. In our case, BookNLP and OWTO use tags, that do not explicitly indicate the beginnings or ends of entities. It is essential to note that there is a tagging format, that uses the prefix "I-" for all cases apart from those where there are two consecutive entities. In that case the "B-" prefix is used to separate the second one from the first. As previously described, due to the differences in the labeling formats, we use the phrase-based approach.

However, there are also different approaches to the evaluation, such as the token-based evaluation, in which the evaluation is done token-by-token. We refrain from this approach due to the fact that not all of our gold standards (i.e. OWTO) and one of the tools (i.e. BookNLP) use prefixes. Unfortunately, this decision also leads to cases, in which it is not possible to objectively determine whether a tool has tagged an entity correctly or not. One example of this is displayed in Table 4.7 a). In this case, the gold standard has two entities "Miss" and "Amelia Sedley". The tool recognises all three tokens as correct, however we do not have a clear indication of whether the annotators detected one or multiple entities within the three tokens. Therefore, a

token-based comparison, would detect the matching of the "I-PER" label as correct, but the other two labels as incorrect. In the alternative case of using the phrase-based evaluation, the CoNLL script[6] also views the labeling of the tool as incorrect. This is due to the fact that the gold standard views two entities (i.e. phrases), while the tool has not clearly distinguished between the tokens and as such has tagged only one entity (i.e. "Miss Amelia Sedley"). Following the approach of using the phrases, the tagging is incorrect.

In an alternative case using prefixes, a possible scenario is presented in Table 4.7 b). Here the honorific "Miss" is tagged as one entity, and the entire name "Amelia Sedley" is marked as a separate entity. In this case, we can state that the gold standard expects two entities and both of them have been tagged correctly by the tool. However, the same three tokens may also be tagged differently by a tool as shown in Table 4.7 c). Here, the spans of the entities (i.e. phrases) are different. Therefore, neither of the entities tagged by the tool are correct.

In summary, this means that the selection of different approaches in the individual steps of the process may lead to different results of the evaluation. We recognise this as one of the major drawbacks of using formats for NER tags, which do not at least include an explicit indicator for the beginning of a new entity. An example of the difference made by the use of prefixes in the evaluation using Flair can be seen in the Appendix D.

[6] `https://www.clips.uantwerpen.be/conll2003/ner/bin/conlleval`

Table 4.7: Determining the Correctness of Entity Recognition

original word	**tool tag**	**gold standard**
a) no indication of beginning of entity		
presenting	O	O
Miss	I-PER	B-PER
Amelia	I-PER	B-PER
Sedley	I-PER	I-PER
to	O	O
b) given span of entity - two correct tags		
presenting	O	O
Miss	B-PER	B-PER
Amelia	B-PER	B-PER
Sedley	I-PER	I-PER
to	O	O
c) given span of entity - two wrong tags		
presenting	O	O
Miss	B-PER	B-PER
Amelia	I-PER	B-PER
Sedley	B-PER	I-PER
to	O	O

Chapter 5

Discussion on Gold Standard Characteristics for NER

Throughout the process of setting up the annotation environment, executing the experiment, and analysing of the evaluation results, many challenges and questions arise. This chapter discusses various limitations and problems we recognised in the field of NER related to data availability, standards, annotation guidelines, challenges of annotating, and evaluation.

5.1 Using Existing Gold Standards for the Literary Domain

Some of the best known and most frequently used corpora for English originate from the MUC-6 task [26], the CoNLL-2003 task [61], and the ACE-2005 conference [67], all of which contain between 30,000 and 400,000 annotated tokens [33]. Further, the CoNLL-2011 shared task uses the OntoNotes corpus, which at that point in time consisted of 1.3 million words. The creators of the task did not make use of the slightly smaller collection of ACE datasets from 2000 to 2004, consisting of 1 million words, due to the fact that the corpus handles few entities and has a lower inter-annotator agreement score [53].

The predominant types of texts annotated in all of these datasets are news or web articles, and conversations. This poses the question of whether these datasets are suitable for the domain of English novels. Rösiger et al. [54] explore the topic of coreference for literary text and state that *"literary texts differ from news texts and dialogues to a great extent, as their purpose is not to transfer information as it is the principle task of a newspaper, but rather to provide poetic descriptions and good storytelling"*. Their statement is supported by detailed research on the structure of literary texts. Overall literary

language shows one clear pattern. It *"tends to use a larger set of syntactic constructions than the language of non-literary novels"* [64]. Furthermore, it applies a mix of direct and indirect speech, and uses rich vocabulary [54].

Taking into account these differences it is essential to consider the individual standards and annotation guidelines used for datasets. Choosing to use the annotated corpus of the CoNLL-2003 shared task [61] might be suitable for the evaluation of a tool, written for NER in news or web articles. However, a tool such as BookNLP, which has been tailormade for the literary domain may not be as good at handling such texts. Instead it may have been trained on literary texts and thus may perform better when recognising entities in novels. In the words of Pradhan et al. *"limitations in the size and scope of the available datasets have also constrained research progress"* [53].

Consequently, using gold standards, which are made for different domains (e.g. news and web articles), as a benchmark for the performance of tools in the context of literature may not yield reliable results. It is essential to avoid tuning tools for the literary domain in a manner, which would improve their performance in different domains (e.g. news articles) as their approach in itself contradicts the purpose of creating domain-specific tools. One example of why this may be counterproductive is the handling of honorifics in old novels. In the novel "David Copperfield" for example, the mother - Clara Copperfield - is addressed as "Mrs. David Copperfield" by a visitor. Annotation guidelines such as CoNLL-2003 do not include honorifics in the span of an entity of type person. In the case of coreference resolution for example this approach may lead to "Mrs. David Copperfield" being recognised as a reference to the narrator (i.e. David Copperfield) or the father (i.e. David Copperfield Sr), instead of as a reference to Clara Copperfield.

A greater focus on the differences and similarities between literary texts and other text types (e.g. historical letters) could be useful in detecting whether or not existing annotated datasets could be properly utilised for the purpose of evaluating or even training existing tools. Such analysis might help in the aim to create *"more consistent annotation of larger amounts of board coverage data for training better automatic techniques for entity and event identification"* [53].

5.2 Maintaining Standards

Many findings about the differences between the types of texts point towards the need for more specified annotated datasets as previously discussed by Rösiger et al. [54]. However, the use of such domain specific datasets would only be beneficial, if their annotation is also adapted to the style of the texts

and the purposes of the tools, being trained and evaluated using them. This makes the creation of more and more datasets a double-edged sword. On the one side tailor made standards could contribute to the progress of NER tools in specific domains. On the other side, the creation of more and specific standards in NER might at a certain point become complex and hinder the possibility of benchmarking tools across multiple domains.

New standards are typically created in order to cover shortcomings of existing ones or to cover fields of study not yet explored. This is noticeable even if we only take as an example the datasets from MUC-6 [26], CoNLL-2003 [61], ACE-2005 [67], OntoNotes 4.0 [70], and CoNLL-2011 [53]. The number of entity types, the sizes of the datasets, and their formats had already evolved in that short period of time between those events. Nowadays, the selection is wider and includes examples such as LitBank, which aim to cover a new domain, yet do not follow the annotation guidelines of any of the bigger existing datasets without adaptations. Projects such as the creation of LitBank could clearly be beneficial for a specific purpose. However, examples from the past show that frequent changes in standards might also be harmful to the targeted progress [53]. The ACE corpus, for instance, had applied iterative adaptations of the task definition and selection of evaluation dataset over many years. This resulted in its complexity, making it difficult to follow all different versions of guidelines and to interpret the performance results objectively, as those are measured following different guidelines over the years [53].

Using old datasets is a stable and possible solution due to the fact that no changes are to be considered. One well known case for this are the CoNLL-2003 guidelines and gold standard, which are still used as of today. It makes evaluations comparable and easier to execute, and may reduce certain bias related to the creation of new gold standards (e.g. interpretation of guidelines, different annotation formats), which we observed throughout the annotation process of the two new gold standards. However, this also comes with the disadvantage that new tools are often tuned to perform better with the datasets they would be evaluated with rather than with newer and more representative gold standards, which cover new progress in the field of NER [46]. Continuously updating all annotation guidelines and the annotated datasets, or creating new datasets may sound like an appropriate solution, yet it is not a trivial task. It oftentimes requires domain knowledge, and multiple annotators, who annotate texts in several steps until they have agreed upon a final version [24].

An alternative solution would be to analyse the differences and similarities between the guidelines and the purposes of said guidelines, and to create bigger clusters of corpora, consisting of similar enough datasets. Further,

one could analyse whether or not the individual collections of datasets could be shaped into individual homogeneous corpora. Our evaluation shows that a random combination of datasets for the evaluation of any tools is not an appropriate solution. Therefore, grouping existing datasets could only be done based on their precise characteristics. For instance, it is beneficial to compare the definitions of the individual entity types (e.g. person), the purposes of the datasets (e.g. conversational networks), the text lengths (e.g. short messages, book chapters), and the languages of the used texts (e.g multi-lingual). In some instances it may be the case that the datasets differ in the entity types they include. Then it may be possible to expand some of the datasets within a corpus to cover more entities and thus the same ones as the other datasets within the same corpus. In other cases it might be sufficient to adapt some annotation formats (e.g. IOT) to fit the rest of the datasets. A detailed study of existing datasets would be beneficial for the creation of a better understanding of the state of the art, and for the exploration of the abovementioned ideas.

5.3 Evaluation Metrics for Tool Performance

With respect to evaluation it is also essential to choose the correct evaluation metrics for the purpose of each analysis. The CoNLL standard, which we followed in this thesis, accepts only full matches between tool tagging and the gold standard as correct. As such, partial correctness is viewed as a mistake. There are a few aspects of this approach to be considered.

First, there are certain use cases, which may not require an exact match in order for a tag to be counted as correct. Let us consider the purpose of detecting whether an entity is mentioned within a sentence. In the field of bioinformatics, the goal may be to *"determine whether or not a particular sentence mentions a specific gene and its function"* [46]. In such cases the span of the entity is less relevant than its presence.

Next, there are some cases, in which it might be more objective to differentiate between "ambiguous" and "incorrect" tags. In the case of single-layered annotated datasets the same token may be a part of multiple entities, forcing annotators to decide which one to include in the final gold standard. This issue is resolved by the use of multiple layers. At other times, it may be unclear from the length and context of the text, what a certain entity refers to due to the fact that the same tokens are sometimes used for various types of entities (e.g. Sofia could be a city and a person's name). When it comes to coreference the ambiguity is more noticeable. Poesio and Artstein [51] give an example from anaphoric annotation, which shows a case, in which

"judgments may disagree - but this doesn't mean that the annotation scheme is faulty; only that what is being said is genuinely ambiguous" [51]:

```
18.1 S:  ....
18.6     it turns out that the boxcar at Elmira
18.7     has a bad wheel
18.8     and they're ..  gonna start fixing that at midnight
18.9     but it won't be ready until 8
19.1 M:  oh what a pain in the butt
```

In this instance, it is unclear whether the token *that* in 18.8 refers to *the boxcar* or to *a bad wheel*. The authors propose to distinguish between cases, in which the annotators cannot come to an agreement over the correct version because of ambiguity, from cases, in which some annotators might have simply made a mistake in the tagging. Multiple correct answers could, for example, be represented by a set of answers [51].

Lastly, our experiment shows how a small difference in the definition of an entity type (e.g. handling of pronouns) between the tool and the gold standard is amplified by the frequency of its occurrence. Thus, the frequent use of pronouns, for example, could drastically reduce the evaluated performance of certain tools in comparison to other tools. Additionally, not tagging certain entities such as animals (e.g. the White Rabbit) could lead to a chain-reaction, in which related entities such as abbreviations (e.g. the Rabbit) are also not tagged. Inevitably, whenever the annotation guidelines of the tool do not fully match the gold standard used for the evaluation of their performance, the evaluation approach is influenced by the frequency of occurrence of the individual rules within the annotation guidelines.

5.4 Selection of Annotation Approach

In certain cases, for example when a domain is not covered by existing gold standards, one might decide to create a new annotated dataset [7, 15, 63]. Independent from the domain, letting multiple people annotate the same texts improves the quality of the final result. As observed throughout our annotation process, this could be due to small human mistakes such as marking the wrong span of an entity, misunderstanding the annotation guidelines, or even due to edge cases not being covered by the set rules. Ideally, more than two annotators should first annotate the same texts and later on reach a consensus over the correct final version. An even number might make the final decision difficult in case that both annotators are convinced that their

decision is the correct one. We observed this case twice throughout the final discussion between our two annotators. In some cases a more insisting annotator might even unintentionally push their opinion irrespective of its correctness. By having three or more annotators involved in the process, disagreements could be solved with a majority vote, and ambiguous cases (i.e. where multiple tags are correct) could also be recognised and agreed upon easier. Lastly, we recognised that our annotators can solve disagreements easier when justifying decisions by using rules from the annotation guidelines. This helps them indicate and avoid intuitive answers, which might not always be in compliance with the guidelines.

An alternative approach to the traditional annotation process, which our annotators would have preferred[1] is annotating as a team. Similar approaches have previously been adopted by other domains such as software programming, where pair programming is used to allow two programmers to work alongside [76]. Applying a similar method in annotation could allow the annotators to clear uncertainties throughout the process and discuss differences in their understanding of the individual guideline rules prior to tagging entire texts. In our experiment the annotators first annotated the texts individually and then met to agree on a final version. Within the final meeting they resolved the majority of the inconsistencies between the two annotations already in the beginning, while discussing the first novel. Those inconsistencies were then reoccurring throughout the different texts. It is important to note that the problem is not that the annotators do not understand the text, the problem is that they do not know how to tag some tokens. Counterintuitively, both annotators found the annotation guidelines to be insufficient despite their length, as many edge cases are not considered in them. In Alice in Wonderland, for example, the annotators agree that "the White Rabbit" is a person entity, as they recognise it as an animal name (see Appendix B, PERSON label). However, the annotators were insecure, whether the tokens "a rabbit" can be seen as referring to the character or just an animal, as the capitalization was not used in this case. Another discussion concerned possessive pronouns. One of the annotators skipped all pronouns followed by a noun (e.g. her saucer), as they viewed them as not referring directly to an existing entity. Simultaneously, the other annotator compared the use of pronouns in this context to the use of an entity's name (e.g. John). In the example "John's saucer", the guidelines say that "John" should be tagged, and therefore this annotator also believed that the pronoun "her" in "her saucer" should be tagged. Therefore, detailed rules are needed to guide annotators

[1]This observation is based on a feedback session done with both annotators after the completion of the annotation process.

and prevent bias.

One further annotation strategy is used by Elson et al. [19] in the context of creating a conversational network using the interactions of characters. In their case, the authors use Amazon's Mechanical Turk program to run a survey. They provide a list of characters from the novel and let three annotators select, which character the speech belongs to. Crowdsourcing is a solution frequently used for the collection of data. One such example is the Galaxy Zoo project [21], which aimed to perform classification over galaxy images. For this purpose, around half a million volunteers contributed by identifying certain features of a galaxy such as its shape (e.g. elliptical) and its spiral category (e.g. "clockwise"). The authors of the paper recognise that the volunteers experienced the feeling of fulfillment through partaking in the project. In some cases their involvement even led to new discoveries [38]. This example shows how crowdsourcing could positively and successfully contribute to the progress in a specific domain. Unfortunately, this is not always the case. A recent study by Northcutt et al. [49] analysed *"10 of the most commonly-used computer vision, natural language, and audio datasets"* [49] and found out that the error in the datasets ranged from 0.15% to 10.12%, with an average of 3.42%. While 3.42% does not sound like much, it is important to consider that the currently best performing NER tool based on CoNLL-2003 - LUKE [74] has an F_1 score of 94.3%, which is merely 0.8% higher than the second best performing tool[2].

5.5 Challenges in Annotation

As observed throughout our annotation process, there are many potential reasons for errors in the annotations. Those may range from insufficient annotation guidelines to inexperienced annotators. In the literary domain for example, some novels are easier to annotate, which can be recognised from the higher inter-annotator agreement score. If we would decide to only use texts with higher inter-annotator agreement for the creation of gold standards, we would limit our selection to simple texts and make it easier for the tools to perform better. By reaching a certain level of representatives in a domain, gold standards could be beneficial for the progress of NER.

An aspect relevant for the literary domain is the length of the texts used. Sometimes short datasets might have too few and repetitive entities (e.g. the novel Dracula in the section of OWTO [15] used for our experiment). In such cases longer datasets might offer a better chance for a tool to be evaluated.

[2] `https://nlpprogress.com/english/named_entity_recognition.html`
Git Commit: 4668fbb454bec04e6182265c1650ba70b0af2aec

From the perspective of an annotator, however, longer texts are difficult to annotate and introduce a higher risk of inconsistency [53]. Amongst other things, longer texts give space to more entities spanned over different parts of the texts, which might hinder the annotators when trying to keep an overview of the entities.

When it comes to the experience of the annotators, it is not always the case that new gold standards are created by experts. Instead annotators are frequently made familiar with the problem setting, and introduced to the task and its guidelines. This may lead to a different errors, which could be grouped together as occurring due to the lack of experience. In the process of annotating, there are two main types of knowledge that annotators apply - text knowledge and world knowledge [54]. Text knowledge refers to the knowledge, which can be found within the text. Annotators could tag entities based on their knowledge as readers or based on the knowledge a character in the book has at the respective point of the story. In our experiment, both annotators used their readers' knowledge. The second type of knowledge - world knowledge - denotes the knowledge that a typical reader would have had at the time of writing of the novel. The lack of this knowledge could influence the labeling decisions of the annotators, as they might not be aware of the fact that certain tokens refer to an entity.

Furthermore, inexperienced annotators may tend to mark tokens, which sounds intuitive to them in a certain way, irrespective of the correctness of their decision. This is especially the case when annotation guidelines do not cover all cases, which we found in the raw texts. We found that it is beneficial for the annotators to first practice annotating on a separate raw text, as this helps them familiarise themselves with the guidelines and even the annotation tool.

Due to the recognised shortcomings of existing datasets, attempts have been made to automatically improve imperfect annotations. CrossWeight is one such framework, which, as described by Wang et al. [69], aims to detect and *"handle label mistakes during NER model training"* [69]. The solution of the authors is driven by their findings that widely-used gold standard datasets such as CoNLL-2003 hold many errors. The authors were able to detect label mistakes in about 5.38% of the test sentences. Considering the currently best F_1 score of 94.3%, a correction of the dataset might certainly lead to changes in the results. Despite the fact that attempts to improve the quality of annotated datasets such as CrossWeight can be successful, reducing the number of errors in the first place could overall increase the level of reliability of the datasets.

5.6 Training Tools on Annotated Datasets

The importance of annotated datasets is relevant not only for the evaluation of tools, but also for training purposes. Due to the complexity of the annotation process and the lack of availability of big annotated datasets in the field of English literature, it is likely that most tools are developed or improved using existing data from other domains. This fact once again underlines the importance of correct labeling.

Over the years, there have been different approaches for the handling of datasets containing erroneous labels such as using algorithms less sensitive to noise, and improving data quality prior to using it [22]. Despite their success in certain aspects, some strategies might negatively influence other aspects of the training sets. For example, filtering labels that seem noisy based on robust loss[3] might unintentionally also filter out labels, which are more difficult to detect. Simultaneously, some wrong labels might also be similar enough to correct ones, making them indistinguishable for automated tools [14]. More work could be done in targeting recognised shortcomings of existing datasets instead of creating new ones in order to derive more precise annotated datasets. This could be done via the help of tools such as CrossWeight [69], but also by letting multiple human annotators find, discuss, and correct faulty labels.

[3] *"Loss correction approaches usually add a regularization or modify the network probabilities to penalize less the low confident predictions, which may be related to noisy samples"* [14].

Chapter 6

Conclusion

In this thesis, we discuss the state of the NER tools and datasets used in the field of English literary novels. In particular, we focus on the named entity type person. We show that the use of different annotation guidelines for the annotation of the same text leads to different gold standards. For this purpose, we use two existing works in the literary domain, which contributed to the creation of datasets, containing annotated English novels - LitBank [6, 7, 57] and OWTO [15]. We first extract the overlapping annotated sections of novels from the two datasets. Then we annotate those overlapping novel sections using the annotation guidelines of CoNLL-2003 [61] and LitBank [6, 7, 57]. Next, we analyse the differences between the evaluated performance of the tools resulting from the use of various gold standards. Lastly, we discuss limitations and problems we recognised in the field of NER.

To answer our first research question *"Which annotated datasets are suitable for the evaluation of off-the-shelf tools for Named Entity Recognition in English literature?"* we looked at the existing annotated datasets in the English literature such as LitBank [6, 7, 57] and OWTO [15]. Alongside these, we considered research that focuses on the differences between literary text and other types of text [54, 64]. We conclude that, due to the specific structures of literary texts as described by Cranenburgh et al. [64], it is more reliable to use domain-specific gold standards for the evaluation of NER tools. We suggest that future work should look at the similarities to closely related domains (e.g. historical letters). A better understanding of the linguistic properties of related domains could help define how they can be used together to create more and better gold standards.

Next, we addressed our second research question *"How does the use of different gold standard datasets created for the domain of English literature affect the measured performance of Named Entity Recognition tools?"* by

using NER gold standards from the literary domain. When used for the evaluation of NER tools, the individual gold standards yield different and oftentimes opposite results in terms of precision, recall, and F_1 score. This makes the evaluation process biased even within the same domain (i.e. literary texts), as the intentional selection of a specific gold standard could lead to better evaluation results for a certain tool.

In order to address our third research question *"What characteristics of a gold standard dataset should be considered when evaluating the performance of Named Entity Recognition tools?"* we analyse the existing datasets and the results of the tool evaluation. Considering the different results yielded by the use of the four gold standards, we identify a need for agreed-upon annotation guidelines to be used for the annotation of literary novels. By using the same guidelines, individual datasets could be combined into a big corpus for the domain and as such could be used for the training and evaluation of tools targeting that domain. This would not only address the need for data as previously discussed by Rösinger et al. [54], but would also allow for the development of unified evaluation processes including the same evaluation metrics (e.g. precision, recall, F_1 score) and potentially the use of a unique evaluation script. Furthermore, we discuss the difference between the use of different prefixes for the tagging of named entities. We identify the use of more explicit formats such as BIO (i.e. beginning, inside, outside) and BIOES/IOBES (i.e. beginning, inside, outside, end, single) as more suitable for an objective evaluation of NER tools. This is due to the fact that the beginning and end of each entity is clearly marked and consecutive entities can be detected as such. In contrast, we find the neglect of prefixes or the use of a single prefix (e.g. I-) to be the most conflicting in terms of the evaluation, as the exact span of individual entities is not given. By neglecting the prefix, which identifies the beginning of an entity, we risk detecting consecutive entities as a single one. One approach that aims to avoid this issue is comparing the tool's output with the gold standard token-by-token (instead of entity-by-entity) and in the process ignoring any existing prefixes in either the gold standard or the tool's output. However, this approach has certain downsides such as tolerating partial correctness and as such not being comparable to evaluation approaches such as the one used in CoNLL-2003 [61], which evaluate the performance entity-by-entity approach and do not accept partial correctness. Therefore, we suggest the use of a format such as BIO or BIOES/IOBES for the creation of datasets in the domain. Lastly, we identify the annotation process as essential for the quality of the gold standard. We base this finding on the observation of our two annotators. Throughout the final meeting of the annotation process, in which both annotators had to agree on the final version of the gold standard, both noticed that they had

unintentionally left out certain entities or had set the span of entities wrong. By letting both of them annotate the same texts and agree on one version in the end, we reduced the number of unintentional human errors in the process of the annotation.

Future work looking at the theoretical aspects of NER in English literature could focus on the following:

- Previous research [54] identifies the need for domain-specific annotated datasets for the literary domain. However, it remains unclear, whether annotated datasets from other related domains might be suitable for the training of tools in the English literary domain.
- Our work takes a closer look at the specific domain of English literature. A further study could assess, whether the same or similar observations can be made for other languages or even for language-independent tools.
- Further, one may look at the differences in performance of general purpose NER tools such as Flair [4] when they are trained using domain-specific datasets in contrast to their standard models.

Looking at the practical aspects of NER in English literature, a natural continuation of this work includes the following:

- While it may be beneficial to have multiple annotation guidelines based on the purpose of the annotation, this may lead to issues in terms of an objective evaluation as shown by our work. We believe that the creation of detailed domain-specific annotation guidelines could help annotators better understand which entities belong to a specific category and which do not.
- Furthermore, such annotation guidelines could be used for the creation of more and unified training and evaluation datasets for the NER tools in English literature. This may also include the adaptation of existing annotated corpora to correspond to the newly agreed upon guidelines.
- We suggest the adoption of a more explicit labeling format using prefixes in order to prevent bias in evaluations (as shown in the example of BookNLP [8] and gold standards using prefixes) and in the interpretation of the span of entities. For this purpose, tools and datasets that use no prefixes should be adapted to explicitly state the beginning of entities

- The final discussion with our annotators indicated that they would have preferred to annotate as a team. This study could be repeated using four annotators split into two groups. Furthermore, one could study, whether crowdsourcing could be suitable to extend the corpus of annotated datasets in the domain without decreasing the quality of annotation.

- Lastly, a question raised by our study is how should tools that output a flat format (i.e. one layer) be evaluated when the gold standard (e.g. LitBank) uses multiple layers.

Bibliography

[1] Linguistic Data Consortium - ACE (Automatic Content Extraction) English Annotation Guidelines for Entities. `https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-entities-guidelines-v5.6.6.pdf`. Last accessed: 2021-03-04.

[2] OntoNotes Release 5.0. `https://catalog.ldc.upenn.edu/docs/LDC2013T19/OntoNotes-Release-5.0.pdf`. Last accessed: 2021-04-12.

[3] Rodrigo Agerri and German Rigau. Robust multilingual named entity recognition with shallow semi-supervised features. *Artificial Intelligence*, 238:63–82, 2016.

[4] Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. Flair: An easy-to-use framework for state-of-the-art nlp. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, 2019.

[5] Alan Akbik, Duncan Blythe, and Roland Vollgraf. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, 2018.

[6] David Bamman, Olivia Lewke, and Anya Mansoor. An annotated dataset of coreference in English literature. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 44–54. European Language Resources Association, May 2020.

[7] David Bamman, Sejal Popat, and Sheng Shen. An annotated dataset of literary entities. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2138–2144. Association for Computational Linguistics, June 2019.

[8] David Bamman, Ted Underwood, and Noah A Smith. A bayesian mixed effects model of literary character. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 370–379, 2014.

[9] Alex Brandsen, Suzan Verberne, Milco Wansleeben, and Karsten Lambers. Creating a dataset for named entity recognition in the archaeology domain. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4573–4577, 2020.

[10] Julian Brooke, Adam Hammond, and Timothy Baldwin. Bootstrapped text-level named entity recognition for literature. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 344–350, 2016.

[11] Ahmad C Bukhari and Yong-Gi Kim. Ontology-assisted automatic precise information extractor for visually impaired inhabitants. *Artificial Intelligence Review*, 38(1):9–24, 2012.

[12] Nancy Chinchor, Erica Brown, Lisa Ferro, and Patty Robinson. Named entity recognition task definition. *Mitre and SAIC*, 1999.

[13] Peter Corbett, Colin Batchelor, and Simone Teufel. Annotation of chemical named entities. In *Biological, translational, and clinical language processing*, pages 57–64, 2007.

[14] Filipe R Cordeiro and Gustavo Carneiro. A survey on deep learning with noisy labels: How to train your model when you cannot trust on the annotations? In *2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 9–16. IEEE, 2020.

[15] Niels Dekker, Tobias Kuhn, and Marieke van Erp. Evaluating named entity recognition tools for extracting social networks from novels. *PeerJ Computer Science*, 5:e189, 2019.

[16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019.

[17] George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie M Strassel, and Ralph M Weischedel. The automatic content extraction (ace) program-tasks, data, and evaluation. In *Lrec*, volume 2, pages 837–840, 2004.

[18] Asif Ekbal, Rejwanul Haque, Amitava Das, Venkateswarlu Poka, and Sivaji Bandyopadhyay. Language independent named entity recognition in indian languages. In *Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages*, 2008.

[19] David Elson, Nicholas Dames, and Kathleen McKeown. Extracting social networks from literary fiction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 138–147, 2010.

[20] Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics, 2005.

[21] Lucy Fortson, Karen Masters, and Robert Nichol. Galaxy zoo. *Advances in Machine Learning and Data Mining for Astronomy*, 2012:213–236, 2012.

[22] Benoît Frénay and Michel Verleysen. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25(5):845–869, 2013.

[23] Adam Geitgey. Natural language processing is fun! *Medium, July*, 18, 2018.

[24] Archana Goyal, Vishal Gupta, and Manish Kumar. Recent named entity recognition and classification techniques: a systematic review. *Computer Science Review*, 29:21–43, 2018.

[25] Shirley Gregor and Alan R Hevner. Introduction to the special issue on design science. *Information Systems & e-Business Management*, 9(1), 2011.

[26] Ralph Grishman and Beth M Sundheim. Message understanding conference-6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*, 1996.

[27] Alan Hevner and Samir Chatterjee. Design science research in information systems. In *Design research in information systems*, pages 9–22. Springer, 2010.

[28] Lynette Hirschman and Nancy Chinchor. Appendix F: MUC-7 coreference task definition (version 3.0). In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*, 1998.

[29] Matthew B Hoy. Alexa, Siri, Cortana, and more: an introduction to voice assistants. *Medical Reference Services Quarterly*, 37(1):81–88, 2018.

[30] Ioana Hulpus, Conor Hayes, Marcel Karnstedt, and Derek Greene. Unsupervised graph-based topic labelling using dbpedia. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, pages 465–474, 2013.

[31] Peter Jackson and Isabelle Moulinier. *Natural language processing for online applications: Text retrieval, extraction and categorization*, volume 5. John Benjamins Publishing Company, 2002.

[32] Andrew Johnson, Penny Karanasou, Judith Gaspers, and Dietrich Klakow. Cross-lingual transfer learning for japanese named entity recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers)*, pages 182–189, 2019.

[33] Markus Krug. *Techniques for the Automatic Extraction of Character Networks in German Historic Novels*. Doctoral thesis, Universität Würzburg, 2020.

[34] Markus Krug, Isabella Reger, Fotis Jannidis, Lukas Weimer, Nathalie Madarász, and Frank Puppe. Overcoming data sparsity for relation detection in german novels. In *DH*, 2017.

[35] Vincent Labatut and Xavier Bost. Extraction and analysis of fictional character networks: A survey. *ACM Comput. Surv.*, 52(5), September 2019.

[36] Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. A survey on deep learning for named entity recognition. *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[37] Elizabeth D Liddy. Natural language processing. *In Encyclopedia of Library and Information Science, 2nd Ed. NY. Marcel Decker, Inc.*, 2001.

[38] Chris J Lintott, Kevin Schawinski, William Keel, Hanny Van Arkel, Nicola Bennert, Edward Edmondson, Daniel Thomas, Daniel JB Smith, Peter D Herbert, Matt J Jarvis, et al. Galaxy Zoo: "Hanny's Voorwerp", a quasar light echo? *Monthly Notices of the Royal Astronomical Society*, 399(1):129–140, 2009.

[39] Edward Loper and Steven Bird. NLTK: The Natural Language Toolkit. *arXiv preprint cs/0205028*, 2002.

[40] Sunghwan Mac Kim and Steve Cassidy. Finding names in trove: named entity recognition for australian historical newspapers. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 57–65, 2015.

[41] Ralph Weischedel Eduard Hovy Mitchell Marcus, Martha Palmer, Robert Belvin Sameer Pradhan Lance Ramshaw, and Nianwen Xue. Ontonotes: A large training corpus for enhanced processing.

[42] Lluis Marquez, Lluis Padro, and Horacio Rodriguez. A machine learning approach to pos tagging. *Machine Learning*, 39(1):59–91, 2000.

[43] Diana Maynard, Kalina Bontcheva, and Isabelle Augenstein. Natural language processing for the semantic web. *Synthesis Lectures on the Semantic Web: Theory and Technology*, 15, 2017.

[44] Mary L McHugh. Interrater reliability: The Kappa statistic. *Biochemia medica*, 22(3):276–282, 2012.

[45] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in Neural Information Processing Systems*, 26:3111–3119, 2013.

[46] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[47] Prakash M Nadkarni, Lucila Ohno-Machado, and Wendy W Chapman. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*, 18(5):544–551, 09 2011.

[48] Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülşen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135, 2007.

[49] Curtis G Northcutt, Anish Athalye, and Jonas Mueller. Pervasive label errors in test sets destabilize machine learning benchmarks. *arXiv preprint arXiv:2103.14749*, 2021.

[50] Ken Peffers, Tuure Tuunanen, Marcus A Rothenberger, and Samir Chatterjee. A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77, 2007.

[51] Massimo Poesio and Ron Artstein. The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 76–83, 2005.

[52] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. Towards robust linguistic analysis using ontonotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152, 2013.

[53] Sameer Pradhan, Lance Ramshaw, Mitch Marcus, Martha Palmer, Ralph Weischedel, and Nianwen Xue. Conll-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–27, 2011.

[54] Ina Roesiger, Sarah Schulz, and Nils Reiter. Towards coreference for literary text: Analyzing domain-specific phenomena. In *Proceedings of the Second Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 129–138. Association for Computational Linguistics, August 2018.

[55] Stefan Schweter and Alan Akbik. Flert: Document-level features for named entity recognition. *arXiv preprint arXiv:2011.06993*, 2020.

[56] Eszter Simon. *Approaches to Hungarian Named Entity Recognition*. PhD thesis, Budapest University of Technology and Economics Budapest, 2013.

[57] Matthew Sims, Jong Ho Park, and David Bamman. Literary event detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3623–3634. Association for Computational Linguistics, July 2019.

[58] Levon Stepanyan. Automated custom named entity recognition and disambiguation. *International Journal of Signal Processing*, 5, 2020.

[59] Jannik Strötgen, Trung-Kien Tran, Annemarie Friedrich, Dragan Milchevski, Federico Tomazic, Anika Marusczyk, Heike Adel, Daria Stepanova, Felix Hildebrand, and Evgeny Kharlamov. Towards the bosch materials science knowledge base. In *ISWC Satellites*, pages 323–324, 2019.

[60] BBN Technologies. Co-reference guidelines for english ontonotes version 7.0. `https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-coreference-guidelines.pdf`, 2007. Last accessed: 2021-04-18.

[61] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147, 2003.

[62] Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics, 2003.

[63] Hardik Vala, David Jurgens, Andrew Piper, and Derek Ruths. Mr. bennet, his coachman, and the archbishop walk into a bar but only one of them gets recognized: On the difficulty of detecting characters in literary texts. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 769–774. Association for Computational Linguistics, September 2015.

[64] Andreas van Cranenburgh and Rens Bod. A data-oriented model of literary language. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1228–1238, 2017.

[65] Jens EL Van Gysel, Meagan Vigus, Pavlina Kalm, Sook-kyung Lee, Michael Regan, and William Croft. Cross-linguistic semantic annotation: Reconciling the language-specific and the universal. In *Proceedings of the First International Workshop on Designing Meaning Representations*, pages 1–14, 2019.

[66] Dániel Varga and Eszter Simon. Hungarian named entity recognition with a maximum entropy approach. *Acta Cybernetica*, 18(2):293–301, 2007.

[67] Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. Ace 2005 multilingual training corpus. *Linguistic Data Consortium, Philadelphia*, 57:45, 2006.

[68] Zihan Wang, Jingbo Shang, Liyuan Liu, Lihao Lu, Jiacheng Liu, and Jiawei Han. CrossWeigh: Training named entity tagger from imperfect annotations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5154–5163. Association for Computational Linguistics, November 2019.

[69] Zihan Wang, Jingbo Shang, Liyuan Liu, Lihao Lu, Jiacheng Liu, and Jiawei Han. Crossweigh: Training named entity tagger from imperfect annotations. In *Proc. 2019 Conferenc. on Empirical Methods in Natural Language Processing and the 9th International Joint Conf. on Natural Language Processing, EMNLP-IJCNLP 2019,*, volume 1, 2019.

[70] Ralph Weischedel, Sameer Pradhan, Lance Ramshaw, Martha Palmer, Nianwen Xue, Mitchell Marcus, Ann Taylor, Craig Greenberg, Eduard Hovy, Robert Belvin, et al. Ontonotes release 4.0. *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium*, 2011.

[71] Yan Wen, Cong Fan, Geng Chen, Xin Chen, and Ming Chen. A survey on named entity recognition. In *International Conference in Communications, Signal Processing, and Systems*, pages 1803–1810. Springer, 2019.

[72] Vanessa Woldenga-Racine. *Issues in Named Entity Recognition on Early Modern English Letters*. Master thesis, University of Washington, 2019.

[73] Vikas Yadav and Steven Bethard. A survey on recent advances in named entity recognition from deep learning models. *arXiv preprint arXiv:1910.11470*, 2019.

[74] Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. Luke: Deep contextualized entity representations with entity-aware self-attention. In *EMNLP*, 2020.

[75] Ganggao Zhu and Carlos A Iglesias. Exploiting semantic similarity for named entity disambiguation in knowledge graphs. *Expert Systems with Applications*, 101:8–24, 2018.

[76] Franz Zieris and Lutz Prechelt. Does pair programming pay off? In *Rethinking productivity in software engineering*, pages 251–259. Springer, 2019.

Appendix A

Differences in Overlapping Sections between Datasets

Due to the fact that LitBank and OWTO did not use the same raw texts for the creation of their annotated datasets. There are some small differences between them. Some of them are due to the use of different versions of the same novels (e.g. American vs. British), others originate from encoding errors. All changes that we have made are noted bellow. No changes were made in the novels *Dracula, Emma, Frankenstein, Oliver Twist, Pride and Prejudice, Vanity Fair.*

Alice in Wonderland:

- encoding corrections in OWTO
  MaÃ¢â‚¬â„¢*am*→ Ma'am

David Copperfield:

- encoding corrections in OWTO
  oÃ¢â‚¬â„¢*clock*→ oâ€™clock

Adventures of Huckleberry Finn:

- encoding corrections in OWTO
  sumf Ã¢â‚¬â„¢*n*→ sumf'n
  more Ã¢â‚¬â„¢*n*→ more'n

Moby Dick:

- period removed from OWTO, does not exist in LitBank
  Loomings O
  . O ←
  Call O
  removed . O

- extra comma removed from LitBank, does not exist in OWTO
  alleys B-FAC
  , O
  streets B-FAC
  and O
  avenues B-FAC
  , O ←
  - - O
  north O

- missing - - entity after Water in LitBank. Removed from OWTO
  ? O
  - - O
  Water O ←
  there O
  is O
  not O
  a O
  drop O

- removed hyphen in token in OWTO
  their O
  huge O
  bakehouses O ←
  the O pyramids O

- added hyphen in token in OWTO
  plumb O
  down O
  into O
  the O
  fore O ←
  - O ←
  castle O ←

- two extra tokens removed from Litbank
  some O
  looking O
  over O
  the O
  bulwarks O
  glasses O ←

! O ←
of O
ships O
from O
China B-GPE

The Call of the Wild:

- encoding corrected in OWTO
  deliver ‘ m → deliver’m
  choke ‘ m → choke’m
  cure ‘ m → cure’m
  takin ‘ m → takin’m

Ulysses:

- extra token removed from LitBank
  you O
  ... O
  . O ←
  He O
  broke O

Appendix B

Annotation Guidelines

The following annotation guidelines have been used for the creation of the two new annotated datasets. In general, the PERSON label follows the CoNLL-2003 annotation guidelines [12], and the PERX label extends the PERSON label by adding the annotation guidelines used for the creation of LitBank [6, 7, 57]. The guidelines for both labels are split into the sections "include" and "ignore", which define what should be tagged as an entity and respectively what shouldn't be.

PERSON Label The following annotation guidelines are directly taken from the MUC-7 annotation guidelines[1][12].

Include:

- A conjoined multi-name expression, in which there is elision of the head of one conjunct, should be marked up as separate expressions." *Bill* and *Susan Jones*
- Treat "possessive forms, e.g., "California's" as multiple tokens [California and 's], unless there is a name such as "McDonald's [burger company]" that is inherently possessive"
- "Proper names used as modifiers in complex NPs are to be tagged when it is clear to the annotator from context or the annotator;s world knowledge that the modifier name is that of" a person (e.g. the *Clinton* government)
- "In a possessive construction, the possessor and possessed (...) substrings should be tagged separately." (e.g. *John*'s *son*)
- Acronyms (e.g. *JS*, when it stands for John Smith)

[1] `https://web.archive.org/web/20060211040221/https://www.nist.gov/speech/tests/ie-er/er_99/doc/ne99_taskdef_v1_4.pdf`

- Nicknames (e.g. "*Mr. Fix-It* [nickname for candidate for head of the CIA]"
- "Quotes are included in the tag if they appear within an entity's name" (e.g. *Vito "The Godfather" Corleone*, but also "*vito the godfather corleone*" and "*vito corleone*, known as *the godfather*") (NOT "if they bound the name" e.g. "*Corleone*, also known as "The Godfather", was the victim of ...")
- "When a definite article is commonly associated with an entity name, it also must be tagged." (e.g. "when *The Godfather* ordered the hit")
- "appositives such as "Jr.," "Sr." and "III" are considered part of a person name" (e.g. *John Doe, Jr.*)
- "Family names are to be tagged as PERSON." (e.g. "the *Kennedy* family" or the *Kennedys*)
- "Animal names are to be tagged as PERSON." (e.g. "*Buddy*, the current president's dog, went to the vet today.")
- "Although religious titles or specifiers such as "saint," "prophet," "imam," or "archangel" are [NOT to] be tagged, the proper name [WILL] be tagged" (e.g. "St. *Christopher* is the patron of")
- References to "God" will be taken to be the "name" of this entity for tagging purposes." (CAUTION: "If it is used as a descriptor, rather than a name, it will not be tagged." e.g. "if you believe in *god* you must...")
- "Names of fictional characters are to be tagged" (e.g. "*batman* has become a popular icon") (CAUTION: "character names used as TV show titles will not be tagged when they refer to the show, rather than the character name" e.g. "adam west's costume from batman the TV series")
- "Fictional animals are a specific type of fictional character, and as such should be tagged" (e.g. *morris the cat* or "that famous advertising icon *speedy*")
- "Tag all instances of entities even when they are repeated either for emphasis or correction" (e.g. "think *dole dole* is tough at all")
  (CAUTION: "If a fragment of a word or entity name occurs at either the beginning or the end of a complete entity name, the fragment will be left out of the tagged name")

Ignore:

- "No nested expressions will be marked" (e.g. "8:24 a.m. Chicago time" is of type TIME only)
- "In some cases, taggable multi-word strings will contain entity name substrings; such multi-word strings are not decomposable; therefore, the substrings are not to be tagged." (e.g. "Arthur Anderson Consulting" is a type ORGANIZATION and there should be "no markup for "Arthur Anderson" alone")
- "Common nouns, including pronouns, used in anaphoric reference to taggable entity names,such as" " "IBM announced that the company would lay off ..." [no markup for "the company"] "
- "Aliases that refer to broad industrial sectors, political power centers, etc." (e.g. "Uncle Sam")
- "Quotes if they bound the entity's name" (e.g. "Corleone, also known as "The Godfather," was the victim of ...")
  (CAUTION: mark, "if they appear within an entity's name" e.g. Vito "The Godfather" Corleone)
- "Non-taggables named after persons should not have the personâ€™s name marked." (e.g. "the movie "Shakespeare's Sister" " [no markup of the person "Shakespeare"])
- "Figures of speech include expressions such as metaphors or similes or devices such as personification or hyperbole." (e.g. "the mark fuhrman of corporate america")
- "Titles such as "Mr." and role names such as "President" are not considered part of a person name. " (e.g. Mr. *Harry Schearer*, mister *bettelheim*)
  (CAUTION: not to be mistaken with "appositives such as "Jr.," "Sr." and "III")
- "Although religious titles or specifiers such as "saint," "prophet," "imam," or "archangel" are [NOT to] be tagged, the proper name [WILL] be tagged" (e.g. "St. *Christopher* is the patron of")
- "References to "God" ", if "used as a descriptor, rather than a name, (...) will not be tagged."
  (CAUTION: regular references to "God" should be tagged)

- "character names used as TV show titles will not be tagged when they refer to the show, rather than the character name" (e.g. "adam west's costume from batman the TV series")
  (CAUTION: regular name mentions "of fictional characters are to be tagged" e.g. "*batman* has become a popular icon")
- "individuals identified by their political affiliation" (e.g. "The Republican stepped into the voting booth.")
- "laws named after people" (e.g. "the Gramm-Rudman amendment")
- "diseases/ prizes named after people" (e.g. "Alzheimer's", "the Nobel Prize")
- "court cases named after people" (e.g. "in the case of joe castano versus the tobacco growers of america")
  (CAUTION: not to be mistaken with mentions of "the person involved in the lawsuit, [in] the lawsuit itself" e.g. "in the *castano* suit, attorneys argued")
- "weather formations" (e.g. "tropical storm arthur")
- "Punctuation marks and special characters are normally considered separate tokens" (e.g. "*Eaton-Sumitomo* joint venture") (however *F. Gregory Fitz-Gerald* is one entity)
- "Apostrophes in transcribed speech (...) found in the case of possessive constructions" (e.g. "at the end of president *bush*'s administration" or "in addition to the *riadys*' donations")

PERX Label The following annotation rules are taken from the annotation guidelines of LitBank [6, 7, 57]. David Bamman's "LitBank Coref Annotation Guidelines" state that they *"generally follow the guidelines set out in Co-reference Guidelines for English OntoNotes"* [2] [60].

Include:

- Personal pronouns that refer to people (e.g. "*He* was a noble man")
- Copular structures[3] (e.g. *John* is *a linguist*)
- Quantifiers (e.g. anybody, somebody, few girls)
- Negated pronouns (e.g. no man, none of us)
- Entire noun phrases describing an entity (e.g. *John, her friend, who she went to school with*, was standing at the door.)
- If a personâ€™s name is introduced by a qualifying nominal (e.g. "uncle Charles") you should treat the nominal and name as a single proper entity (in other words, tagging the whole span of "uncle Charles" as a PERX).

Ignore:

- Base plurals - *People* need to breathe.
- Exclamations - *Jesus Christ!* What happened here?
- Pleonastic - *It* was the only way, things could have been done.
- Figurative language - "*The young man* was not really *a poet*; but surely he was a poem. [Note: "poem" is not tagged here]"[4]

[2] https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-coreference-guidelines.pdf

[3] "A copular structure consists of a referent (usually the subject), an attribute of that referent (usually the predicate), and a copula that serves to equate (or link) the referent with the attribute. In the example below, [John] is the referent, [a linguist] is the attribute, and "is" is the copula."[60]

[4] Example provided by David Bamman

Appendix C

Technical Notes

C.1 BookNLP Output Format

A generic example of the command used to call the pipeline of BookNLP is given in Figure C.1. The input file it requires is the raw text of the novel. The tool outputs a file, containing one token per line.

Figure C.1: Example of BookNLP Usage

```
./runjava novels/BookNLP -doc /path/to/original.txt -p /path/to/
    diagnostics -tok /path/to/example.tokens -f
```

The file, which is important for the purpose of this work is the one containing the tokens. It contains the following columns[1]:

- paragraphId - the id of a paragraph, starting from 0
- sentenceID - the id of a sentence, starting from 0
- tokenId - the id of a token, starting from 0
- beginOffset - first character of the token
- endOffset - last character of the token
- whitespaceAfter - indication whether there is a white space after a token (not the case for words including hyphens for example)
- headTokenId - syntactic head id (-1 for the sentence root)

[1]column names extracted from an actual output file, descriptions expanded using the provided description `https://github.com/dbamman/book-nlp`

- originalWord - original token
- normalizedWord - normalized token
- lemma - lemma of the token
- pos - Penn Treebank POS tag
- ner - NER tag (PERSON, NUMBER, DATE, DURATION, MISC, TIME, LOCATION, ORDINAL, MONEY, ORGANIZATION, SET, O)
- deprel - Stanford basic dependency label
- inQuotation - Quotation label (begin quote, inside quote, outside quote)
- characterId - same for all coreferent tokens
- supersense[2]

[2] `https://wordnet.princeton.edu/documentation/lexnames5wn`

C.2 Comparing LitBank Versions

We compared a previously created GitHub repository for LitBank[3] to the latest repository[4]. The comparison approach can be seen in Figure C.2. This simple approach found no differences between the files.

Figure C.2: Comparing the "old" and the "new" LitBank Repositories

```
mkdir /mnt/data/old_litbank
cd /mnt/data/old_litbank
git clone https://github.com/dbamman/NAACL2019-literary-entities.git
    .
for filename in /mnt/data/litbank/entities/tsv/*.tsv; do
    bookname=$(basename -- "$filename")
    diff /mnt/data/litbank/entities/tsv/$bookname /mnt/data/
        old_litbank/litbank/entities/tsv/$bookname
done
```

[3] https://github.com/dbamman/NAACL2019-literary-entities
[4] https://github.com/dbamman/litbank

C.3 Tagging Litbank Raw Texts using BookNLP

The raw texts provided in the collection of novels Litbank are the full texts of the novels. We run the algorithm on these texts. Due to their size and the fact that BookNLP is programmed as a single-threaded tool, the processing of the entire text took ca. 10 hours[5]. The following approaches could be used to reduce this time:

- On average about one chapter of the book or ca. 2000 words per novel have been annotated. However, during the evaluation of the performance of the tool we detected that the raw text provided for the evaluation of tools (e.g. BookNLP) is longer. Therefore, for the comparison we cropped the compared lines of raw text to correspond to the length of the provided annotation. One approach to speed up the process would be to reduce the text length in advance instead of after the tagging of the entire raw texts via BookNLP. In order to make replication of this part of the process faster, we provide the output files of BookNLP in the repository. Note that for a replication those files might need to be moved to the correct folder, as the existing paths in the scripts are written correctly for a complete replication.

- Another approach to speed up the process would be to adjust the java process to run in a multi-threaded manner. We did not follow this approach in order to keep the replication as simple as possible.

- The last and probably easiest solution to the slowness of the tagging is to split the books into multiple folders and tag the individual subsets from separate containers. We used this approach for the tagging of the second set of data by Dekker et al [15]. We split the files equally into three sub-folders and started three docker containers in parallel. This allowed us to process the books faster, however the memory usage accordingly peaked up to 7.2 GB. This factor may need to be taken into consideration, if the steps are to be replicated.

[5]If the process is ran on a separate node, it is recommendable to run the step in a tmux shell in order to avoid the risk of a disconnection in between.

C.4 Flair Models and Configuration

At the time of writing, the tool supports a selection of 8 word and document embeddings[6]. Additionally, the user can decide to combine any of those in a so called "StackedEmbeddings" class. Furthermore, one may chose to use embeddings on a document level instead of word level.

For the training of the model, Flair facilitates a simple setup for accessing publicly available datasets for NLP. Based on the annotation guidelines, the task and the targeted language(s) one can select from nine corpora. The dataset is then downloaded and automatically split into training, testing and development sections.

Further, if one prefers using the tool without a specific training, Flair *"includes a model zoo of pre-trained sequence labeling, text classification and language models"* [4]. The pre-trained models are distributed in two variants - "default" and "fast". The default variants require a GPU to be run on and use embeddings with 2048 hidden layers. The fast variants can be ran with a simpler setup using CPU and use embeddings with 1024 hidden layers. The selection of trained sequence tagger models spans over 16 models for English, 4 multilingual models, 10 models for German, and 10 models for other languages, covers 11 tasks using 17 different datasets[7]. For the purpose of NER in English the currently best performing model is "ner-large"[8], which scores an F_1 score of 94.09 with the CoNLL-2003 training dataset.

Sample code on how we use Flair is shown in Figure C.3. It uses several code snippets provided in the documentation of the tool in its GitHub repository. It our case we read the entire text to be tagged as a segment and use the available splitter to split it into a list of sentences. The individual sentences are then passed to the "predict" method for the actual tagging.

[6] classic word embeddings, hierarchical character features, byte-pair embeddings, character-level LM embeddings (i.e. Flair), pooled version of Flair, word-level LM embeddings (i.e. ELMo), ELMo transformer, and byte-pair masked LM embeddings (i.e. Bert)

[7] `https://github.com/flairNLP/flair/blob/master/resources/docs/TUTORIAL_2_TAGGING.md` Commit: daa1c02868ebd908cc605cd8bfa0c84b4e050e28

[8] `https://huggingface.co/flair/ner-english-large`

Figure C.3: Example of Flair Usage for NER

```
from flair.models import SequenceTagger
from flair.tokenization import SegtokSentenceSplitter

# load text from file
path = "/mnt/data/gold_standard/overlap/original_texts/sample.txt"
with open(path,'r') as file:
    text = file.read()

# load model
tagger = SequenceTagger.load('ner-large')

# initialize sentence splitter
splitter = SegtokSentenceSplitter()
# use splitter to split text into list of sentences
sentences = splitter.split(text)

# predict tags for sentences
tagger.predict(sentences)

# print all tokens with their ner tag and confidence in the
   prediction
for sentence in sentences:
    for token in sentence:
        tag = token.get_tag('ner')
        print(f'{token} {tag.value} {round(tag.score,2)}')
```

Appendix D

Evaluation of Flair on Token Level using Prefixes

It is not the goal of this thesis to compare different evaluation approaches. However, we believe that it is important to keep in mind that the selection of the evaluation approach may lead to different results. In addition, it may diverge from the aim for an objective evaluation of the performance of tools using certain standards (e.g. CoNLL-2003). Table D.1 shows the results of a token-based evaluation of the tool Flair in contrast to a phrase-base evaluation as shown in the main body of the thesis. Note that we could not evaluate Flair using the OWTO gold standard following this approach, as it uses a single prefix (i.e. I-). A simple comparison of the phrase-based and token-based evaluations shows that the calculated performance is different.

Table D.1: Token-based Evaluation of Flair

Novel	LitBank			OWTO			New (CoNLL)			New (Ext)		
	P	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**	**P**	**R**	**F_1**
Alice in Wonderland	76.92	21.05	33.06	x	x	x	100.00	78.79	**88.14**	100.00	11.30	20.31
David Copperfield	93.55	10.51	**18.89**	x	x	x	6.45	12.50	8.51	6.45	0.47	0.88
Dracula	62.50	3.52	6.67	x	x	x	62.50	22.73	**33.33**	62.50	1.23	2.42
Emma	70.69	14.14	23.56	x	x	x	55.17	65.31	**59.81**	55.17	7.42	13.09
Frankenstein	53.33	3.00	5.67	x	x	x	60.00	81.82	**69.23**	60.00	1.96	3.79
The A. of Huckleberry Finn	89.47	27.20	41.72	x	x	x	73.68	73.68	**73.68**	78.95	9.29	16.62
Moby Dick	90.00	3.80	7.29	x	x	x	90.00	100.00	**94.74**	70.00	1.93	3.75
Oliver Twist	76.19	4.78	8.99	x	x	x	90.48	90.48	**90.48**	85.71	4.16	7.93
Pride and Prejudice	31.37	6.67	11.00	x	x	x	94.12	100.00	**96.97**	88.24	10.44	18.67
The Call of the Wild	91.43	12.70	22.30	x	x	x	88.57	64.58	**74.70**	91.43	8.86	16.16
Ulysses	92.68	30.89	46.34	x	x	x	92.68	92.68	**92.68**	92.68	18.91	31.40
Vanity Fair	65.04	18.52	28.83	x	x	x	48.78	65.22	**55.81**	38.21	9.59	15.33
Mean	74.43	13.07	21.19	x	x	x	71.87	70.65	**69.84**	69.11	7.13	10.84
Standard deviation	19.09	9.57	14.02	x	x	x	26.96	27.92	27.08	26.88	5.42	9.16
Median	76.56	11.61	20.6	x	x	x	81.13	76.24	**74.19**	74.48	8.14	13.09