KTH ROYAL INSTITUTE
OF TECHNOLOGY

Degree Project in Computer Science and Engineering

Second cycle, 30 credits

Rule-based coreference resolution for German using morphological and semantic information

CAROLINE GUSTAFSSON

Stockholm, Sweden, 2024

Rule-based coreference resolution for German using morphological and semantic information

CAROLINE GUSTAFSSON

Degree Programme in Computer Science and Engineering
Date: May 20, 2024

Supervisor: Johan Boye
Examiner: Viggo Kann
School of Electrical Engineering and Computer Science
Swedish title: Regelbaserad koreferenslösning för tyska med hjälp av morfologisk och semantisk information

Abstract

Coreference resolution is an area within natural language processing, namely the task of determining which expressions in a text refer to the same entity. It is a useful resource for other natural languages processing activities, such as text summarization. Although the latest methods within the field of coreference resolution utilize deep learning, earlier rule-based methods can still be relevant in certain cases.

One of the rule-based coreference resolution models developed for German is CoRefGer-rule. It implements the Stanford sieve algorithm originally developed for other languages, such as English. The authors reported that CoRefGer-rule was useful especially for out-of-domain texts where no gold standard information was available, arguing the relevancy of rule-based models. They also gave improvement suggestions for their model. One of the suggestion was to add a morphological analyzer in order to make use of grammatical information about words. Another suggestion was to implement a rule that considers the semantic aspect of words, i.e. the aspect related to the meaning of words.

In this project, we developed our own rule-based coreference resolution model based on CoRefGer-rule, where these two suggestions were examined. One of our conclusions was that the use of grammatical information obtained from the morphological analyzer was a fundamental and important part of our system, enabling advanced resolution of pronouns. On the other hand, the semantic information had no significant impact on the performance of our model.

Keywords

Coreference resolution, natural language processing, German

Sammanfattning

Koreferenslösning är ett språkteknologiskt område som handlar om att avgöra vilka uttryck i en text som refererar till samma fenomen i världen. Koreferenslösning kan i sin tur användas för andra språkteknologiska syften, t.ex. automatisk textsammanfattning. De senaste metoderna för koreferenslösning använder sig av djupinlärning, men även äldre regelbaserade metoder kan vara lämpliga för användning i vissa fall.

En av de regelbaserade modellerna som utvecklats för tyska är CoRefGerrule. Den bygger på den s.k. Stanford sieve-algoritmen som ursprungligen utvecklades för ett antal andra språk, bl.a. engelska. Utvecklarna av CoRefGerrule menade att deras modell var användbar i synnerhet för texter utanför domänen som saknade guldstandardsdata. De gav även förslag på hur deras modell skulle kunna förbättras. Ett av förslagen var att bygga in morfologisk analys för att kunna utnyttja grammatisk information om ord. Ett annat förslag var att lägga till en regel som tar hänsyn till ordsemantik, alltså ordens betydelse.

Det här projektet gick ut på att utveckla ett system inspirerat av CoRefGerrule och inkludera de funktioner som nämndes i förbättringsförslagen ovan. En av våra slutsatser var att den grammatiska information som extraherades från den morfologiska analysen var en fundamental och viktig del av vårt system. Den möjliggjorde bl.a. avancerad koreferenslösning för pronomen. Samtidigt visade sig den semantiska informationen inte ha någon särskild påverkan på resultatet.

Nyckelord

Koreferenslösning, språkteknologi, tyska

Acknowledgments

I would like to thank my supervisor Johan Boye for his support throughout the project. I would also like to thank the University of Tübingen for providing me with the TüBa-D/Z corpus and the GermaNet database.

Stockholm, May 2024
Caroline Gustafsson

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Coreference resolution is an area within natural language processing, namely the task of determining which expressions in a text refer to the same entity [1]. For example, let us consider the following sample text:

> Anna has a dog. She likes to play fetch with it.

Now, we highlight all the expressions referring to real-life entities:

> [Anna] has [a dog]. [She] likes to play fetch with [it].

As we can see, the expressions referring to real-life entities in this text are ”Anna”, ”a dog”, ”She”, and ”it”. Now, we traverse the text left-to-right in order to determine which entities these expressions refer to, and in particular, if any expressions refer to the same entity. The first expression is ”Anna”, which refers to a specific person. The second expression is ”a dog”, which refers to a specific animal. The third expression is ”She”, which refers to a specific person – namely, the same person that is referred to by the first expression, ”Anna”. Thus, we say that ”Anna” and ”She” corefer. The last expression is ”it”, which refers to the same entity as ”a dog”. In order words, we have found another coreference chain: ”a dog” and ”it”.

In summary, in this sample text we identified two different entities. The first entity is a specific person, referred to by two expressions: ”Anna” and ”She”. The second entity is a specific animal, also referred to by two expressions: ”a dog” and ”it”. By identifying these entities and their corresponding expressions, we have performed coreference resolution. As we

can see from this example, expressions can refer to the same entity despite the fact that they consist of different words. An entity can also be referred to by a single expression. However, in that case, the expression is not part of any coreference chain.

Coreference resolution is a useful resource for other natural languages processing activities, including text summarization and question answering [1]. The task of identifying expressions and their corresponding entities is usually trivial for a human, but can be challenging for a computer. Throughout the years, different approaches to this task have been suggested. Coreference resolution approaches range from rule-based models, through machine learning models, to deep learning models. Rule-based models make up the earliest approach, whereas neural machine learning models make up the latest [2].

In coreference resolution for the German language, all different approaches have been utilized. What appears to be the current state-of-the-art for German news text is a deep learning model from 2021 [3]. Deep learning being the most recent approach, most models developed for German to date are rule-based or machine learning based. In 2018, a rule-based coreference resolution system named CoRefGer-rule was developed by Srivastava et al. [4]. The system is an adaptation of the Stanford sieve algorithm [5] [6] [1] to German. This approach builds on a succession of rules, called sieves, where the goal of each sieve is to find pairs of mentions that corefer. The original Stanford sieve system has received state-of-the-art results for several text databases and genres [1].

While CoRefGer-rule did not achieve state-of-the-art results for German news texts, the authors concluded that rule-based models are useful especially for out-of-domain texts, where sufficient training data is not available. Thus, they argued the relevancy of such models. They also gave suggestions for improving their system. One of their suggestions was to add a so-called morphological analyzer in order to make use of detailed grammatical information about words, as a lack of this aspect was considered to be one of the main disadvantages of their system. Another suggestion was to add a new so-called semantic sieve that is based on the meaning of words [4]. Here, they referred to the semantic sieve featured in a rule-based coreference resolution model for historic German novels, implemented by Krug et al. [7], and suggested that it could possibly be expanded to other domains.

In this thesis, we will examine these two suggested features. Specifically, we will examine the influence of a morphological analyzer and semantic information on a coreference resolution system based on CoRefGer-rule.

1.2 Research question

How will adding a morphological analyzer and semantic information to a rule-based coreference resolution system for German, based on CoRefGer-rule, influence the performance?

1.3 Thesis structure

The main part of this thesis is divided into five chapters. Chapter 2, Background, provides an overview of linguistic and natural language processing concepts relevant to this project. It also provides a general overview of the field of coreference resolution, in particular regarding the German language, and presents some relevant metrics for evaluating coreference resolution systems. Chapter 3, Method, describes the data and resources used for this project as well as the details of our coreference resolution system. Chapter 4, Results, presents the scores achieved by our system based on the evaluation metrics introduced previously. Some parts of the system are also analyzed in more detail. Furthermore, a comparison of our model to other German coreference resolution models is made. Chapter 5, Discussion, provides reasoning about these results, in particular regarding the contribution of the morphological analyzer and the semantic information to our project. It also contains error analyzes and reflections on ethical and sustainability aspects of this project. Finally, chapter 6, Conclusions and future work, concludes this thesis by summarizing the findings and suggesting ideas for further research.

Chapter 2

Background

2.1 Linguistic and natural language processing concepts

In this section, we will introduce linguistic concepts in order to describe relevant parts of German grammar and other terms used for the coreference resolution system of this project. Example words and expressions will be given in English if applicable, otherwise the German words and expressions will be supplemented with English translations. Because of the project's nature, this section will also cover some natural language processing terms.

Please note that the German grammar and the linguistic concepts introduced here are more complex than it may appear from the basic examples given below. The aim is to provide a general explanation rather than an extensive overview. For more detailed information we refer to German grammar resources [8] [9] [10] and general linguistic resources [11].

2.1.1 Morphology, syntax, and semantics

Linguistics can be further divided into several core areas, such as phonology, morphology, syntax, semantics and pragmatics [9]. In this section, morphology, syntax, and semantics will be covered, as these are the relevant areas for the topic of this report.

Morphology studies the structure of words. Words are built up by so-called morphemes, which are the smallest units of meaning in a word [11]. For example, the word "cats" consists of two morphemes which each has their own meaning: the morpheme "cat" refers to a real-world entity (a cat), and the morpheme "-s" carries a grammatical function (it denotes multiplicity).

There exist different kinds of morphemes. A morpheme which forms the base of a word is called a root, such as ”cat” in the previous example. Further, a morpheme which cannot form a word of its own but must be attached to other morphemes is called an affix [11], such as ”-s” in the previous example.

Syntax concerns the structure of sentences, and regulates how words can be combined to form statements and expressions [11]. For example, the sentence ”The white cat eats a fish.” seems fine. On the other hand, ”*Eat fish a the cat white” seems to make no sense. The reason is that the first sentence follows the syntax rules for the English language, and the second sentence does not. Sentences consist of clauses, which in turn consist of phrases, and syntax concerns the content and combination of such phrases [11]. In German, a clause usually consists of a noun phrase and a verb phrase. Phrases are an essential part of coreference resolution and will be further explained below.

Semantics studies the meaning of linguistic expressions. Meaning exists on different levels [9], but in this report only lexical semantics, i.e. word-level meaning, will be considered. Some words have a single or several very closely related meanings, such as ”giraffe” which refers to ”a large fleet African ruminant mammal [...]” [12]. Other words can have several very distinct meanings, such as ”spring” which among others can refer to ”the season between winter and summer [...]” [13] or ”an elastic body [...]” [13]. There also exist different types of semantic relations. Synonymy refers to linguistic expressions which differ in form but are semantically equivalent*. As an example, ”job” and ”occupation” are synonyms, as they refer to the same phenomenon. Two other semantic relations, hyperonomy and hyponomy, concern hierarchy: a hyperonym is a superior whereas a hyponym is a subordinate [11]. For example, the word ”cat” is a hyperonym of the word ”ragdoll” (a cat breed) and, conversely, ”ragdoll” is a hyponym of ”cat”.

2.1.2 Parts of speech

Parts of speech, also called word-classes, refer to categories of words [11]. There is no single ”true” classification for the parts of speech in German, as the classification depends on what criteria (morphological, syntactical, and semantic) are considered [9]. For this project, the classifications of the Stuttgart-Tübingen-TagSet (STTS) [14] will be used. The reason is that STTS

*Synonyms can be divided into two groups: absolute and partial. Absolute synonyms are expressions that can be exchanged in all possible contexts, wheres partial synonyms are expressions that are very similar but can have slight differences in e.g. connotation or stylistics. Since absolute synonyms are very rare [9], ”synonymy” will refer to expressions of both groups throughout this report.

seem to be commonly used for the annotation of German corpora, and hence relevant in natural language processing tasks for German. In STTS, there are 11 parts of speech: nouns, verbs, articles, adjectives, pronouns, cardinal numbers, adverbs, conjunctions, adpositions, interjections, and particles. The most important parts of speech for entity-based coreference resolution are nouns and pronouns, so these will be covered in detail below. Since articles and adjectives are closely related to nouns and pronouns, they will also be mentioned. Furthermore, verbs will be briefly mentioned because of their syntactical importance in sentences. The remaining parts of speech will not be explained, as this is outside the project's scope.

2.1.2.1 Noun

Nouns can be divided into two groups: proper nouns and common nouns. Proper nouns are names of individuals (e.g. ”Anderson”), geographic locations (e.g. ”Berlin”), companies (e.g. ”Toyota”) etc. Common nouns refer to concrete or abstract things, such as ”cat”, ”situation” or ”love” [8].

In German, all nouns start with a capital letter, unlike English where only proper nouns are capitalized. Furthermore, German nouns are inflected according to three categories: gender, number, and case. Inflection means that the word changes it form based on the grammatical context [11], and German is considered to be a highly inflectional language. The inflection of nouns is called noun declension and an overview of the German noun declension is given in figure 2.1.

- **Gender:** German has three genders, which are made visible through their corresponding articles [9]: masculine, e.g. ”der Hund” (”the dog”), feminine, e.g. ”die Katze” (”the cat”) and neuter, e.g. ”das Pferd” (”the horse”). There exist rules for the determination of a noun's gender that are based on the noun's meaning or form, but in many cases the gender determination is arbitrary [8]. Animate entities can have so-called natural gender, which means that their grammatical gender conforms to their sex. For example, ”die Frau” (”the woman”) is feminine whereas ”der Mann” (”the man”) is masculine.

  However, there are many exceptions to the rule of natural gender, e.g. ”das Mädchen” (”the girl”). In the opposite manner, feminine and masculine forms are, unlike in English, not reserved only for animate entities with natural gender. For example, ”die Tür” (”the door”) is feminine and ”der Tisch” (”the table”) is masculine. All in all, all three genders can be used for animate as well as inanimate beings.

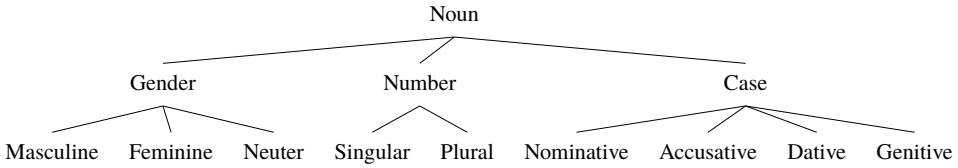

Figure 2.1: Overview of the German noun declension. The graph is a translated and slightly modified version of the graph in Meibauer et al. [9].

- **Number:** German has two numbers, which are made visible through inflectional elements: singular and plural [9]. Singular denotes a single entity, e.g. ”Hund” (”dog”), whereas plural denotes multiple entities, e.g. ”Hunde” (”dogs”).
- **Case:** German has four cases: nominative, accusative, dative, and genitive [9]. Cases are based on syntactical factors, and it is outside the project's scope to describe these in detail. However, it is important to point out that the case influences the form of the article and sometimes also the form of the noun. For example, ”the dog” corresponds to four different German translations based on the case: ”der Hund” (nominative), ”den Hund” (accusative), ”dem Hund” (dative), and ”des Hund(e)s” (genitive).

2.1.2.2 Article

In German, articles accompany nouns with the purpose of showing gender, number and case as they are inflected accordingly, like in the examples above. Articles can be definite (”der”, ”die”, ”das” in singular nominative) or indefinite (”ein”, ”ein”, ”eine” in singular nominative) [9]. Generally speaking, a definite article refers to a specific entity which is known among the speakers, e.g. ”der Hund” (”the dog”). An indefinite article, on the other hand, may refer to an entity which is introduced by the speaker for the first time, such as ”ein Hund” (”a dog”) [10]. An indefinite article can also be used to refer to a general phenomenon, where the speaker does not have a specific entity in mind.

2.1.2.3 Adjective

An adjective is a word that describes a noun [10], such as ”brown”, ”small” and ”amazing”. Like the articles, adjectives accompany nouns and are inflected

according to gender, number and case in German. As an example, the adjective ”braun” (”brown”) becomes ”braune” in the expression ”der braune Hund” (”the brown dog”).

2.1.2.4 Pronoun

Pronouns are often used as substitutes for nouns and the words connected with them. However, there exist different kinds of pronouns, which differ in function and inflection. Here, the pronouns relevant for this report are listed and explained briefly: personal pronouns, reflexive pronouns, possessive pronouns, and relative pronouns.

- **Personal pronouns:** In German, personal pronouns are inflected according to number and case. They also have an additional category, which is called person. There are three persons: 1st, e.g. ”ich” (”I”), 2nd, e.g. ”du” (”you”), and 3rd, e.g. ”es” (”it”). 3rd person singular is also inflected according to gender. The 3d person singular nominative personal pronouns are ”er” (masculine), ”sie” (feminine) and ”es” (neuter) [10].

  Animate entities with natural genders get the pronouns that correspond to their sex, such as ”die Frau” – ”sie” (”the woman” – ”she”) and ”der Mann” – ”er” (”the man” – ”he”). This includes the nouns that are exceptions to the rule of natural genders, e.g. ”das Mädchen” – ”sie” (”the girl” – ”she”). All other pronouns get the genders of the nouns that they substitute regardless of animacy, such as ”der Tisch” – ”er” (”the table” – ”it”). Thus, unlike in English, masculine, feminine and neuter personal pronouns can all refer to animate as well as inanimate entities in German.

- **Reflexive pronouns:** A reflexive pronoun refers back to a noun in the same clause. For example, in the sentence ”I know myself very well”, ”myself” is a reflexive pronoun. In German, reflexive pronouns are inflected according to number and person [10].

- **Possessive pronouns:** A possessive pronoun refers to an owner. For example, in the sentence ”She has lost her bag”, ”her” is a possessive pronoun. In German, possessive pronouns are inflected according to gender, number, case and person [10].

- **Relative pronouns:** A relative pronoun refers back to a noun in a previous clause. For example, in the incomplete sentence ”The man,

who has a dog, ...", "who" is a relative pronoun. In German, relative pronouns are inflected according to gender, number and case [10].

2.1.2.5 Verb

A verb denotes an action, a change or a state [10]. For example, in the sentence "It rains", "rains" is a verb, and in the sentence "She has lost her bag", "has" and "lost" are both verbs.

2.1.3 Phrases

A phrase is a syntactic unit that consists of one or several words [11]. As mentioned in a previous section, phrases are the constituents that make up clauses. A phrase has a so-called head, which can be seen as its core or minimally required element [8]. Though there are different kinds of phrases, we will focus on explaining noun phrases, because of their key role in the kind of coreference resolution that is applied in this project. Verb phrases will also be briefly mentioned because of their syntactical relevance.

2.1.3.1 Noun phrase

The head of a noun phrase is a noun or a pronoun. Further words can be added to the phrase, and these are called determiners and modifiers*. Determiners come before the head and consist of articles or pronouns, whereas modifiers can come before or after the head, and can consist of various other phrases [8] [15]. For example, in the phrase "the brown dog in the park", we have the following constituents: "dog" is the head, "the" is a determiner, and "brown" and "in the park" are modifiers.

2.1.3.2 Verb phrase

A verb phrase has a verb as its head. There exist different definitions of the term "verb phrase", and we will refer to the definition in Jurafsky and Martin [16] because of the way that the verb phrase is applied in the next section about syntax trees. According to this definition, a verb phrase consists of a verb (or several) and may in addition contain other phrases. For example, in

*Different terms exist that refer to the parts of a noun phrase that are not head or determiner. For instance, they can be divided into further categories: "premodifier", "complement" and "postmodifier" [15]. In this report, "modifier" will be used to cover all categories, for simplicity reasons. "Modifier" is also the term used in the Stanford sieve algorithm [1].

the sentence, ”The dog chased the red ball in the park”, everything but the initial noun phrase, ”the dog”, constitutes the verb phrase.

2.1.4 Syntax trees

Sentences can be displayed as tree-like structures in order to show relations between words. In this section, two types of syntax trees will be introduced: constituency trees and dependency trees*.

2.1.4.1 Constituency tree

In a constituency tree, a sentence's phrase-structure is shown. A simple sentence (S) consists of a noun phrase (NP) and a verb phrase (VP) [16]. These phrases are further divided into smaller components – phrases in phrases – until only the individual words remain, which are denoted with their corresponding parts of speech. The syntax tree for the sentence ”The dog chased the red ball in the park” is shown in figure 2.2. For the part of speech tags in this tree, we refer to the STTS tagset [14].

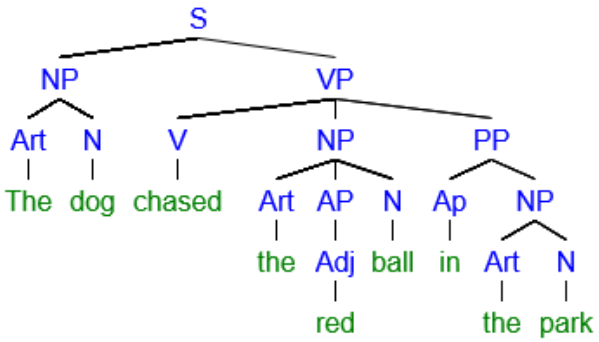

Figure 2.2: The syntax tree for the sentence ”The dog chased the red ball in the park”. PP = prepositional phrase, AP = adjective phrase. The tree was generated using `http://mshang.ca/syntree/`.

2.1.4.2 Dependency tree

Another way to represent the structure of a sentence is through a dependency tree. This tree does not show phrases but rather grammatical relations between words, which go from heads (or headwords) to dependents. Each sentence has

*The terms ”constituency tree” and ”dependency tree” are taken from Lee et al. [17].

a root, which can be seen as the head of the sentence itself. Each word in the sentence has a direct or indirect relation to the root. Dependency relations are labeled with tags that denote their corresponding types of grammatical relations [16]. In figure 2.2, the dependency tree for the sentence ”The dog chased the red ball in the park” is shown. For the relation tags, we refer to the Universal Dependencies [18].

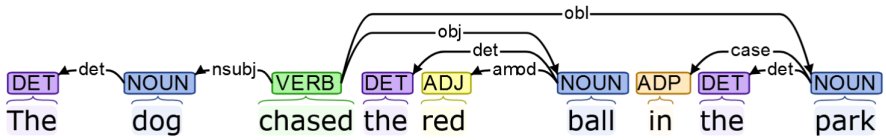

Figure 2.3: The dependency tree for the sentence ”The dog chased the red ball in the park”. The tree was generated using http://stanza.run/.

2.1.5 Other relevant terms

In this section, other linguistic and natural language processing terms relevant for the coreference resolution system of this project are introduced.

2.1.5.1 Lemma

The lemma is the dictionary form of a word, that is, it is a word that does not contain inflectional elements. Rather, it is a representation of all its inflected forms [19]. For example, the lemma of ”dog”, ”dogs” and ”dog’s” is ”dog”. The task of finding the lemma of a word is called lemmatization [16]. Lemmatization will prove to be important for the coreference resolution system of this project. This is because of the extensive German noun declension, which generates many different forms of the same lemma, as shown above.

2.1.5.2 Apposition

An apposition is a noun phrase with the purpose of specifying or describing another noun or pronoun. In German, it is placed directly after this noun or pronoun and is usually separated by commas [8]. For example, in the expression ”Berlin, the capital of Germany”, ”the capital of Germany” is an apposition.

2.1.5.3 Predicate nominative

Predicative nominative refers to a noun phrase in a copulative relation. A copulative relation could be described as an ”equals” relation, and consists of two noun phrases connected by a copula [8]. A copula is usually a verb, and in English a typical copula is ”(to) be”. For example, in the sentence ”She is an engineer”, ”she” and ”an engineer” can be seen as equals, and ”an engineer” is the predicate nominative.

2.1.5.4 Abbreviation

An abbreviation is a word that consists of initial letters from a multi-word combination. There are two kinds of abbreviations: initialisms and acronyms, where the difference is the way they are pronounced [11]. In German, abbreviations can take many forms. Unless otherwise stated, the following examples are all names of German political parties. An abbreviation may consist only of capitalized letters, e.g. ”SPD”, ”Sozialdemokratische Partei Deutschlands”, but it may also contain lowercase letters, e.g. ”AfD”, ”Alternative für Deutschland”. Furthermore, it can contain several initial letters from the same word, such as ”BfTh”, ”Bürger für Thüringen”.

Furthermore, German compound words are not separated by space as can be the case in English, but are written together or separated by hyphens. This can also influence the form of the abbreviation. For example, ”Christlich-Soziale Union in Bayern” is shortened ”CSU” and ”Bundesausbildungsförderungsgesetz” (a regulation on financial assistance for students in Germany) is shortened ”BAföG”. There also exist many other abbreviations whose form differentiates from the examples mentioned.

2.1.5.5 Demonym

A demonym is a noun that denotes an inhabitant in or a person coming from a specific place [20]. For example, an inhabitant in Germany is called ”German” and an inhabitant in Greece is called ”Greek”.

2.1.5.6 Named entity

In the field of natural language processing, a named entity (NE) is an entity that is referred to with one or several proper nouns, such as ”Angela Merkel”. The task of identifying such entities is called named entity recognition (NER). When a named entity has been identified, it is usually provided with a label stating the name category, such as ”PERSON” (e.g. for ”Angela

Merkel"), "LOCATION" (e.g. for "Berlin"), and "ORGANIZATION" (e.g. for "Toyota") [16].

2.1.5.7 Token

Tokenization is the task of segmenting text into smaller parts, such as words, as used in natural language processing. The resulting parts are called tokens. A token is not necessarily a word or an entity separated by a whitespace, as it could also be e.g. a punctuation mark [16]. For example, if we would tokenize the sentence "It rained for a while, then it stopped.", we would get the following tokens: "It", "rained", "for", "a", "while", ",", "then", "it", "stopped", and ".".

2.1.5.8 Parsing

In natural language processing, parsing refers to the task of identifying the structure of sentences, such as identifying the phrase structure of a constituency tree or the dependency relations of a dependency tree. The computer program carrying out this task is called a parser [21]. A parser may also identify other grammatical features of words, such as noun genders. Naturally, tokenization is a necessary part of the parsing process.

A common parsing output format is the CoNLL-U format of the Universal Dependencies [22]. Here, each word is output on its own line, including nine other fields. Each field represents some of the word's features, such as part of speech or dependency relation. In figure 2.4, we can see an example of the CoNLL-U format, where the German word "Katzen" ("cats") has been parsed as part of some imaginary sentence. The fields will not be explored in detail here. However, it should be noted that column 2 denotes the word itself, column 3 denotes its lemma, column 4 denotes its general part of speech, column 6 denotes part of speech-specific grammatical information, and column 8 denotes its dependency relation. For the tags used here, we refer to STTS [14] and Universal Dependencies [18].

```
3 Katzen   Katze   N   NN   Fem|Nom|Pl   4   nsubj   _ _
```

Figure 2.4: The word "Katzen" ("cats") parsed as part of some imaginary sentence and presented in CoNLL-U format.

2.1.5.9 Word vector

In natural language processing, word meaning is most often represented in the form of word vectors, also called word embeddings*. Using this approach, a word is represented by a vector in ”a multidimensional semantic space” [16]. The vector is created based on the word’s co-occurrence with other words in sentences. In order to determine similarity between words, the distance between their vectors is measured. Vectors appearing close to each other in the semantic space represent words with similar meaning, and vice versa for vectors appearing far from each other [16].

A common vector distance metric is cosine similarity. The cosine similarity refers to the cosine of the angle between two word vectors $\vec{u}$ and $\vec{v}$. The metric value ranges from 0 to 1, where 0 is to be interpreted as ”least similar” and 1 is to be interpreted as ”most similar”. Equation 2.1 shows the cosine similarity formula [16].

$$\cos(\vec{u}, \vec{v}) = \frac{\vec{u}\vec{v}}{|\vec{u}||\vec{v}|} \tag{2.1}$$

2.2 Coreference resolution

This section will introduce some relevant terms within coreference resolution, specify what kind of resolution is the focus of this project, explain different resolution algorithms, review recent coreference resolution corpora and models for German and present relevant metrics for evaluation.

2.2.1 Mention and cluster

Two key words in coreference resolution are ”mention” and ”cluster”. A mention is a span of words that refers to an entity. In this project, we will only consider mentions that refer to real-life entities, as described in the next section. Further, a cluster is the set of mentions that refer to the same entity. We will illustrate the meaning of these concepts through an example. Consider the following text:

> Annie has a dog. It likes to play fetch, but she thinks playing fetch is boring. She would rather cuddle with her cat.

*Sometimes the term ”word embedding” is used with a more strict definition than ”word vector” [16]. However, it is beyond the scope of this project to discuss the matter.

Now, we highlight all the mentions, i.e. all word spans that refer to a real-life entity, and denote them with IDs. Each mention has its own unique ID, which is marked with a subscript, as in Lee et al. [1]. The task of identifying mentions in a text is called mention detection.

> [Annie]$_1$ has [a dog]$_2$. [It]$_3$ likes to play fetch, but [she]$_4$ thinks playing fetch is boring. [She]$_5$ would rather cuddle with [[her]$_6$ cat]$_7$.

As we can see, the text contains six mentions. Each mention then belongs to a cluster, and what cluster it belongs to depends on what entity it refers to. We now try to find the clusters by processing the text left-to-right. Mention 1, ”Annie”, has no antecedent. An antecedent is any previous mention in the text, and for obvious reasons, the first mention in a text has no antecedent. For now, ”Annie” is the only mention referring to the ”Annie” entity, and thus ”Annie” can be said to be in its own cluster. Similarly, mention 2, ”a dog”, does not refer to the same entity as any previous mention. Therefore, it is also the only mention in its cluster. Mention 3, ”It”, however, refers to the same entity as the previous mention ”a dog”, and is therefore in the same cluster as ”a dog”. Further, mention 4, ”she”, refers to the same entity, as ”Annie”, and thus ”she” and ”Annie” belong to the same cluster, and so on. Giving each cluster a unique ID and marking it with a superscript as in Lee et al. [1], the text now looks like this:

> [Annie]$_1^1$ has [a dog]$_2^2$. [It]$_3^2$ likes to play fetch, but [she]$_4^1$ thinks playing fetch is boring. [She]$_5^1$ would rather cuddle with [[her]$_6^1$ cat]$_7^3$.

Now, we can see that the text contains seven mentions and three clusters: cluster 1 contains the mentions ”Annie”, ”she”, ”She” and ”her”, cluster 2 contains the mentions ”a dog” and ”It”, and cluster 3 contains the single mention ”her cat”. Mentions appearing alone in their clusters are called singletons.

2.2.2 Resolution types

In the field of coreference resolution, there exist different but closely related resolution applications. Here, we introduce some of these with the purpose of clarifying what kind of resolution will be used for this project.

2.2.2.1 Entity vs. event coreference resolution

Entity coreference resolution is about determining which mentions in a text refer to the same real-life entity [23]. A real-life entity could for example be a person, an animal or a (physical or abstract) object. Thus, the typical mention used here is a noun phrase. Consider the following sentence, where the entity mentions are highlighted:

[The dog]$_1$ chases [a ball]$_2$.

Here, ”the dog” and ”a ball” refer to different real-life entities. Another type of resolution is called event coreference resolution, which is slightly more complex than the entity based type [2]. As the name implies, it focuses on events rather than real-life entities [23]. If we highlight the event mentions in the sentence introduced above, it looks like this:

The dog [chases]$_1$ a ball.

In this project, we will only be concerned with entity coreference resolution, as this is the type of resolution for which the Stanford sieve method was developed [1].

2.2.2.2 Anaphora vs. coreference resolution

The terms ”anaphora resolution” and ”coreference resolution” are often used interchangeably. However, they are not equal but overlapping kinds of resolution [2]. Anaphora is defined as ”use of a grammatical substitute (such as a pronoun or a pro-verb) to refer to the denotation of a preceding word or group of words” [24]. A typical mention in anaphora resolution is indeed a pronoun, as a pronoun is often used to replace an already introduced noun phrase.

Coreference resolution, however, can also cover mentions which do not have an obvious referring form. Thus, coreference resolution mentions that refer to the same entity can differ greatly in ”grammatical structure and function (e.g., gender and part of speech)” [2]. In this project, we will be concerned with coreference rather than anaphora resolution, but as already indicated, this type of resolution can also cover anaphoric mentions.

2.2.3 Coreference resolution algorithms

This section gives a brief overview of three types of coreference resolution models: rule-based models, machine learning models, and deep learning

models. Rule-based models will be reviewed in more detail than the others, as it is the relevant model for this project. The referenced work in this section features models applied to the English language.

2.2.3.1 Rule-based models

Rule-based models rely on rules based on syntactic and semantic features. The rules are often hand-crafted and knowledge-rich, although some models try to minimize the knowledge dependency [2]. An early system is that of Hobbs [25] proposed in 1978, implementing a syntax-based pronoun resolution. In the algorithm, a parse tree is traversed left-to-right and breadth-first in the search of an antecedent. The system utilizes selectional constraints in order to prune the tree until only a single antecedent remains. Since this early system precedes the use of standardized evaluation methods, the algorithm was evaluated manually [2].

Lappin and Leass [26] also implemented a system for pronoun resolution in 1994, but in addition to the syntactic information it also includes other aspects, such as semantics. Futhermore, their system is based on the so-called salience principle. All antecendents have a salience value that is calculated as a sum of their features. The chosen antecendent is then the antecedent with the maximum saliance value. Their system outperformed Hobbs' [2], and several other researchers also based their systems on the salience principle [27].

Another trending algorithm principle was built on the centering theory. A center is defined as an entity that is referred to and links utterances together. The two key rules of the centering theory specify that pronominalizations usually refer to the current center and that it is preferred to keep the same entity as the current center [2]. One of the algorithms utilizing this theory was the BFP algorithm by Brennan, Friedman, and Pollard [28] published in 1987. An advantage of the centering theory was its independence of extra-linguistic semantic information, but a disadvantage was that it preferred references between sentences rather than within a sentence [2].

CogNIAC [29], introduced in 1997, was one of the models developed with the goal of reducing knowledge dependency. So-called knowledge-poor systems use "heuristic methods instead of complete syntactic knowledge or world knowledge" [27] and usually require larger datasets than knowledge-rich systems [27]. In the CogNIAC anaphor resolution algorithm, when an antecedent is matched in accordance with a rule, that antecendent is chosen and no more rules are assessed. This may lead to unresolved anaphors if no rules find an antecedent. The CogNIAC system's performance was in line with

Hobbs' [2].

Haghighi and Klein [30] developed a deterministic rule-based model in 2009. Their system applies a succession of rules, ordered by relevance. Despite its simplicity, it outperformed many of the systems at the time of publishing. The idea of rules ordered by relevance was then extended by other researchers, resulting in the Stanford sieve approach [1], as mentioned in the introduction to this report. The model has seen several improvements over time. It was initially introduced by Raghunathan et al. [5] in 2010 and extended as a part of the CoNLL-2011 shared task [31][6]. In 2013, Lee et al. [1] presented an even more detailed version.

The Stanford sieve approach was motivated by the fact that a single feature-based function for choosing antecedents could lead to errors. This is because low-precision features could risk dominating the choice of antecedent at the expense of high-precision features. Precision refers to the chance of yielding relevant matches as opposed to irrelevant matches*. Therefore, the rules in the Stanford sieve algorithm are ordered so that high-precision rules are applied earlier [2].

The two main stages of the model are mention detection and coreference resolution. In the mention detection stage, mentions are selected on the basis of syntactical criteria. In the coreference resolution stage, a set of ten independent rules, called the "multi-pass sieve", is applied in the order described above. During this stage, mentions are clustered together according to the rules. A unique feature of the model is the entity-centric aspect. This means that when resolving a mention, a candidate antecedent not only shares its own (e.g. syntactical and semantic) features but also those of the mentions in its cluster. The Stanford sieve model achieved state-of-the-art performance [1] and lays the foundation for the method applied in this project. Hence, more details to the model will be presented in chapter 3.

2.2.3.2 Machine learning models

Rule-based models were eventually outperformed by machine learning models. With the emergence of machine learning methods, the coreference resolution research community "expanded from linguists to machine learning enthusiasts" [2]. Unlike rule-based models, rules do not need to be hand-crafted but can be extracted automatically along with features, as long as there is access to sufficient data [27].

There exist different categories of machine learning approaches, such as

*The concept of precision is further described in section 2.2.5.

the mention-pair model, the entity-mention model and the ranking model. The mention-pair model looks at pairs of mentions and antecedents and uses a binary classifier to decide whether or not they belong to the same cluster. The entity-mention model instead makes resolution decisions based on whole clusters, not single mentions. In turn, the ranking model ranks all candidate antecedents – or candididate clusters – and chooses the one with the highest ranking [2].

A relevant model to be mentioned here is the one introduced by Lee et al. [17] in 2017. After identifying strengths and weaknesses of the rule-based Stanford sieve model, they combined this model with statistical methods from the machine learning field of coreference resolution. The precision hierarchy and the modular architecture from the sieve model are retained, while most other aspects are machine learning based. Their system was evaluated on English, and outperformed the corresponding rule-based model.

2.2.3.3 Deep learning models

A disadvantage with the machine learning models is that they are feature rich. Determining what features to use can be difficult, and extracting them can be time consuming. This became less of a problem with deep learning approaches. Deep learning also lead to the development of word vectors, a way to represent the semantic aspect of words [2], as mentioned previously.

The earliest deep learning approaches for coreference resolution used a nonlinear model with a pipeline of preprocessing steps. Later, a so-called end-to-end model was introduced. This model does not need a preprocessing pipeline as it carries out both mention detection and coreference resolution simultaneously, enabling a more efficient process [27].

2.2.4 Coreference resolution for German

In this section, we introduce some relevant corpora and systems developed for coreference resolution in German.

2.2.4.1 Corpora

TüBa-D/Z (Tübinger Baumbank des Deutschen/Zeitungskorpus) [32] is the standard corpus for German coreference resolution [3]. It has seen several releases over the years, and the latest release is version 11 from 2018. Version 11 consists of 3,816 manually annotated articles from the German newspaper taz (die tageszeitung), with a total of 104,787 sentences and 1,959,474 tokens.

In addition to coreference relations, the annotations provide information on inflection, lemmas, syntactical categories, named entities, dependency relations, and more. To use the dataset, a license from the University of Tübingen is required [33].

For the German part of the SemEval-2010 task, as described in the section below, a version of TüBa-D/Z was used. This is referred to as the SemEval-2010 dataset [34]. It consists of 1,235 articles, 26,098 sentences, and 455,046 tokens.

Another German corpus is DIRNDL (Diskurs-Informations-Radio-Nachrichten-Datenbank für Linguistische Analysen) [35] [36] introduced in 2012. It contains German news broadcasts and is annotated with coreference resolution information and speech features. As of 2014, the corpus contained 3,221 sentences and about 50,000 tokens. It is available online*.

DROC (Deutsches Roman Corpus) is a German literature corpus presented in 2018 [37], manually annotated with coreference resolution for novel characters as well as other features relevant for the literature domain. It contains excerpts from 90 novels and the source code is available online†.

For information on some earlier German corpora, we refer to Tuggener [38].

2.2.4.2 Resolution systems

Several coreference resolution systems have been developed for German, the vast majority being rule-based and machine learning based. Here, we will focus on giving a brief review of some of the more recent works, dating from 2010 and forward. For a chronological review of the earlier German coreference resolution, we refer to Tuggener [38].

In 2010, Klenner, Fahrni, and Sennrich [39] developed a machine learning system for German anaphora resolution. Utilizing a morphological analyzer, a NER tool, a dependency parser and a wordnet, they used morphological, syntactic and semantic filters to choose possible antecedents for a mention. They evaluated their model with gold standard‡ information from TüBa-D/Z. One of their conclusions was that the German morphology introduced ambiguities that caused errors during the parsing process. However, they also

*https://www.ims.uni-stuttgart.de/en/research/resources/corpora/dirndl/

†https://gitlab2.informatik.uni-wuerzburg.de/kallimachos/DROC-Release

‡In this report, we use the terms ”gold” and ”gold standard” to refer to the correct labelling of data.

stated that the resulting performance drop was quite small when using a parser compared to when using the gold standard information.

SemEval is a series of natural language processing workshops where the goal is to contribute to the research in semantic analysis and develop relevant datasets within the field [40]. In 2010, the SemEval task 1 was ”coreference resolution in multiple languages” [34]. German was one of six languages and the corresponding corpus used was the SemEval-2010 corpus. Several systems was used for the task and overall, the systems scored the second best results for German. However, a problem that has been identified is that singletons are included in the evaluation, which can bias the result [41].

CorZu, a model for German coreference resolution, was introduced by Klenner and Tuggener [41] in 2011. It has received improvements over time [42] [38] and was last updated in 2016. It is a hybrid incremental entity-mention system, combining both rule-based and machine learning approaches. Furthermore, antecedents for mentions are chosen based on Markov logic networks. In 2016, the CorZu system was evaluated on TüBa-D/Z version 9.1 and provided state-of-the-art results for German pronoun resolution. The model is available online*.

Krug et al. [7] performed coreference resolution within the domain of literature in 2015. Their system is based on the previously mentioned Stanford sieve model. However, the rules were adapted and new sieves were introduced in order to tailor the system to the domain. In particular, the identification of characters in novels was seen as the main objective. The evaluation was done on their own novel samples and the system achieved state-of-the-art results for the German literature domain.

The IMS HotCoref coreference resolution model was developed by Björkelund and Kuhn [43] in 2014 for Arabic, Chinese, and English. In 2016, Rösiger and Kuhn [44] adapted the model to German, naming it IMS HotCoref DE. IMS HotCoref is a data-driven model involving a machine learning algorithm, where the text is represented as a directed tree. For IMS HotCoref DE, no changes were made to the learning algorithm, but features of the system regarding e.g. word form, gender, and animacy were adapted to the German grammar. The system was evalued on the TüBa-D/Z version 10 and SemEval 2010 corpora.

As mentioned in this report’s introduction, in 2018 Srivastava et al. [4] adapted the rule-based Stanford sieve model to German and called the system CoRefGer-rule. However, they also developed a rule-based system for English, machine learning models for German (CoRefGer-stat) and English,

*https://github.com/dtuggener/CorZu

and a projection-based approach for German (CoRefGer-proj). In CoRefGer-proj, they translated German text to English, ran it through a coreference resolution model developed for English, and translated it back to German. The results showed that CoRefGer-rule performed better than their other German models when evaluated on TüBa-D/Z and SemEval-2010.

Another advantage of the rule-based model was its use for out-of-domain texts within the field of digital curation. Digital curation is a multidisciplinary field focusing on the management and preservation of digital resources [45], such as letters and news. The digital curation texts that the authors used were out-of-domain, as these texts did not contain any gold standard information. This means that they could not be used for the training phase of a supervised machine learning model. The authors tested the out-of-domain texts on all their models and their judgment was that the rule-based models achieved the best performance for German as well as English.

CoRefGer-rule applies six out of the seven sieves proposed in the 2010 version of the Stanford sieve approach [5]. As the authors did not have access to a morphological analyzer, thus lacking information about the gender, number, and case of nouns and pronouns, they could not implement the sieve for pronoun resolution. Furthermore, the lack of a morphological analyzer was said to be the reason for the system's underperformance. They also suggested other ways of improvement, such as adding the semantical sieve as implemented by Krug et al. [7]. Since the goal of this project is to build on CoRefGer-rule, the relevant details of their system will be covered in chapter 3. The source code for CoRefGer-rule is available online*.

Krug [46] applied coreference resolution within the literature domain in 2020, focusing only on character references. They used rule-based, machine learning, and deep learning methods. The results showed that the Stanford sieve model outperformed the other models in several metrics, as evaluated on the DROC corpus.

In 2021, Schröder, Hatzel, and Biemann [3] introduced what appears to be one of the first deep learning models for German. They adapted existing approaches and used two different architectures based on the documents' length: a coarse-to-fine architecture for short documents and an incremental architecture for long documents. Their end-to-end model was evaluated on TüBa-D/Z version 10, SemEval-2010 and DROC. The coarse-to-fine approach of the model outperformed IMS HotCoref DE and CorZu, achieving state-of-the-art performance on German news datasets. Their code and models have

*https://github.com/dkt-projekt/e-NLP/tree/master/src/main/java/de/dkt/eservices/ecorenlp/modules

been published online*.

Gupta, Hatzel, and Biemann [47] performed coreference resolution within the domain of literature in 2024. One of their goals was to develop a method for performing resolution on long documents. Their model was evaluated on two novels and outperformed both Krug [46] and Schröder, Hatzel, and Biemann [3].

2.2.5 Evaluation metrics

A so-called F1 score is usually evaluated for both mention detection and the clustering task. F1 score, in turn, relies on the two partial metrics precision (P) and recall (R). However, for the evaluation of the clustering task, the traditionally used formulas for precision and recall are not suitable because of the task's non-binary classification nature. Instead, other metrics have been suggested. The most commonly used are MUC, B^3, CEAF and CoNLL, where CoNLL is the average of the MUC, B^3 and CEAF F1 scores [23]. These will be the metrics used for the evaluation of the clustering task of this project.

2.2.5.1 F1 score

The most common definitions of precision and recall in binary classification tasks are based on positive and negative predictions: true positive predictions (tp), false positive predictions (fp) and false negative predictions (fn). The formulas for precision and recall are shown in equations 2.2 and 2.3 [23], and will be used to evaluate the mention detection part of this project.

$$P = \frac{|tp|}{|tp| + |fp|} \tag{2.2}$$

$$R = \frac{|tp|}{|tp| + |fn|} \tag{2.3}$$

The F1 score then "relates to the harmonic mean of precision and recall" [23] and its formula is shown in equation 2.4 [23]. The same F1 formula is used for MUC, B^3 and CEAF, where P and R refer to their respective calculations for precision and recall, as presented in the following sections.

$$F1 = \frac{2}{P^{-1} + R^{-1}} \tag{2.4}$$

*https://github.com/uhh-lt/neural-coref/tree/konvens

2.2.5.2 MUC

MUC was introduced by Vilain et al. [48] in 1995 and assumes links between mentions. Correspondingly, the calculation of precision is based on a partition function which returns the number of different gold clusters that the mentions in a predicted cluster belong to, and vice versa for recall. For example, if the mentions in some predicted cluster belong to 3 different gold clusters, the partition function returns 3 [23]. The formulas for MUC precision and recall are shown in equations 2.5 and 2.6, where p denotes the predicted clusters, g denotes the gold clusters, and N_p and N_g denote the number of predicted and gold clusters respectively [1].

$$P_{MUC} = \frac{\sum_{i=1}^{N_p}(|p_i| - |\text{partition}(p_i)|)}{\sum(|p_i| - 1)} \tag{2.5}$$

$$R_{MUC} = \frac{\sum_{i=1}^{N_g}(|g_i| - |\text{partition}(g_i)|)}{\sum(|g_i| - 1)} \tag{2.6}$$

2.2.5.3 B^3

B^3 was introduced a few years after MUC by Bagga and Baldwin [49]. It is based on the number of overlapped mentions between clusters. Precision and recall are computed for each mention by calculating the number of mentions that appear both in its gold cluster and its predicted cluster, then dividing by the number of mentions in the gold cluster and predicted cluster respectively. The formulas are shown in equations 2.7 and 2.8, where p_i denotes the mention's predicted cluster and g_i denotes the mention's gold cluster [23].

$$P_{B_i^3} = \frac{|g_i \cap p_i|}{|p_i|} \tag{2.7}$$

$$R_{B_i^3} = \frac{|g_i \cap p_i|}{|g_i|} \tag{2.8}$$

The resulting precision and recall are the sums of the individual mentions' precision and recall. The sums are weighted by $w_i = \frac{1}{N_m}$, where N_m is the number of mentions. These formulas are shown in equations 2.9 and 2.10 [23].

$$P_{B^3} = \sum_{i=1}^{N_m} w_i P_{B_i^3} \tag{2.9}$$

$$R_{B^3} = \sum_{i=1}^{N_m} w_i R_{B_i^3} \tag{2.10}$$

2.2.5.4 CEAF

The CEAF metric was proposed by Luo [50] in 2005. It includes a function ϕ which takes as input a single gold cluster g and a single predicted cluster p, as seen in equation 2.11. Four different versions of ϕ exist, but we only use the so-called entity-based version presented here, because it is the one included in the CEAF F1 metric [23].

$$\phi(g, p) = 2\frac{|g \cap p|}{|g| + |p|} \tag{2.11}$$

The ϕ function contributes to the calculation of precision and recall. For each gold cluster g_i, a mapping to a predicted cluster $\text{km}(g_i)$ is done based on similarity, according to the Kuhn-Munkres algorithm [51]. The gold cluster and the mapping are then input to the ϕ function. The formulas for precision and recall are presented in equations 2.12 and 2.13 below, where N_p and N_g denote the number of predicted and gold clusters respectively [23]. Again, different versions of these formulas exist, but as we use the entity-based version of ϕ, we will use the corresponding entity-based versions of the formulas for precision and recall.

$$P_{CEAF} = \frac{\sum_{i=1}^{N_g} \phi(g_i, \text{km}(g_i))}{N_p} \tag{2.12}$$

$$R_{CEAF} = \frac{\sum_{i=1}^{N_g} \phi(g_i, \text{km}(g_i))}{N_g} \tag{2.13}$$

2.2.5.5 CoNLL F1 score

CoNLL was proposed in 2012 [52] and is simply the average of the MUC, B^3 and CEAF F1 scores. The reason for combining the measures is that they are all somewhat biased towards different aspects of the evaluation of the clustering tasks. The CoNLL F1 formula is shown in equation 2.14 [23].

$$F1_{CoNLL} = \frac{F1_{MUC} + F1_{B^3} + F1_{CEAF}}{3} \tag{2.14}$$

Chapter 3

Method

Although the source code for CoRefGer-rule is available online, we chose to instead develop our own coreference resolution system from scratch and implement the relevant sieves from CoRefGer-rule based on the description by Srivastava et al. [4]. The reason for not using the original code is that CoRefGer-rule is published as part of a bigger natural language processing system (e-NLP). So, we judged it difficult to extract only the coreference system part. Therefore, our system cannot be seen as an extended version of CoRefGer-rule, but rather an independent system heavily influenced by it. While CoRefGer-rule is written in Java, we wrote our program in Python.

3.1 Data and other resources

In this section we will describe the data and the other external resources used in our coreference resolution system.

3.1.1 Corpus: TüBa-D/Z

As corpus we chose the latest version of TüBa-D/Z [32], provided by the University of Tübingen, because it is considered to be the standard corpus for German coreference resolution [3]. Details of this corpus are described in section 2.2.4.1. The test data consisted of 349 articles, 6,767 sentences and 117,281 tokens. Initially, we aimed to use about 200,000 tokens, but due to problems with running the parser on big volumes of data, we had to narrow the number of tokens down.

3.1.2 Parser: ParZu

For the parsing, we used ParZu (The Zurich Dependency Parser for German) [53] [54], which is available for download without a license*. It is a dependency parser and morphological analyzer implemented by the Computational Linguistics Group at the University of Zurich, that outputs the parsed data in CoNLL format. In addition to dependency relations, ParZu provides information about a word's lemma, general part of speech, specific part of speech and part of speech-distinct grammatical features (such as gender, number, and case of a noun).

We also considered two other morphological analyzers: DEMorphy [55] and SMOR (Stuttgart Morphological Analyzer) [56]. However, we chose ParZu because of its availability and rich content (dependency parsing as well as morphological analysis).

3.1.3 NER tool: Stanford CoreNLP

As NER tool we used Stanford CoreNLP [57] for German. It identifies four NER categories: location, person, organization, and misc. Stanford CoreNLP is a natural language processing toolkit, developed by The Stanford NLP Group, that also provides other information, such as part of speech identification and dependency parsing. However, for German, it does not provide lemma identification and part of speech-distinct grammatical features. Therefore, it was not considered as a parser candidate. Stanford CoreNLP is available online†.

We initially tried using another NER tool, GermaNER [58]. Nonetheless, after observing several errors in the GermaNER output, we judged that it's performance was not good enough. Hence, we switched to Stanford CoreNLP.

3.1.4 Lexical-semantic database: GermaNet and germanetpy

For the the semantic sieve suggested by Srivastava et al. [4], as used in Krug et al. [7], we used the latest version (version 18.0) of GermaNet [59] [60] ‡. GermaNet is a license-requiring lexical-semantic database, developed by members of the Division of General and Computational Linguistics at the

*https://github.com/rsennrich/ParZu

†https://stanfordnlp.github.io/CoreNLP/

‡We also used GermaNet as a part of another sieve. See chapter 3 for details.

University of Tübingen. It is inspired by and compatible with the Princeton WordNet [61] for English. GermaNet contains German nouns, adjectives, and verbs and provides semantic relations between them, such as synonymy, hypernymy and hyponymy. It also divides the words into categories. For example, some noun categories are ”Mensch” (”human”), ”Tier” (”animal”) and ”Körper” (”body”). The latest version of GermaNet contains a total of 215,000 words [62]. However, in our system we only used the part containing the nouns.

In order to simplify the implementation and use of GermaNet in our program, such as extracting the semantic relations, we used their API system called germanetpy*.

3.1.5 Word vectors: fastText

In order to add another semantic aspect to our program we used word vectors. We chose pre-trained German word vectors provided by fastText† of dimension 300. We chose these word vectors because of their availability and because we were already familiar with fastText word vectors. However, other pre-trained German word vectors also exist, such as the GloVe vectors and Word2Vec vectors provided by deepset‡.

3.2 The Stanford sieve approach

The architecture of the Stanford sieve approach is divided into two main parts: the mention detection and the sieve algorithm. The purpose of mention detection is to extract all mentions from the corpus that may be relevant for coreference resolution. The mentions extracted from this process are called predicted mentions. In turn, the purpose of the sieve algorithm is to create clusters of mentions. The algorithm takes the mentions as input and iterates through a series of sieves in order to gradually cluster the mentions together.

In this section, we will describe how we performed mention detection, explain the general steps of the sieve algorithm, and present the sieves we used.

*https://pypi.org/project/germanetpy/
†https://fasttext.cc/docs/en/crawl-vectors.html
‡https://www.deepset.ai/german-word-embeddings

3.2.1 Mention detection

The mention detection process described in Lee et al. [1] is based on syntax trees. However, since we used a dependency parser, we had access to dependency trees rather than syntax trees and thus could not directly apply their method. Instead, we referred to the dependency tree-based mention detection process described in a degree project about Swedish coreference resolution by Vällfors [63].

Since the Stanford sieve algorithm is tailored for noun phrases [1], we only considered noun phrases as mentions. As previously mentioned, a noun phrase has a noun or a pronoun as its head. Thus, in the first step to find a mention, we looked for a noun or a pronoun. Then, in accordance with Vällfors [63], we followed the dependency relations from the head in order to get the complete mention. We included all of the words whose head was the head of the phrase itself, or whose head was any of the other words already added to the mention.

We also applied some filters in order to get rid of what appeared to be errors in the mention detection, or remove mentions judged irrelevant for coreference resolution. The filters are listed below.

1. We removed all mentions that appeared as a part of a previous mention. This is because we noticed that in many cases, a single gold mention could correspond to several predicted mentions. For example, the single gold mention ”der Landesvorsitzende Hans Taake” (”the regional chairman Hans Taake”) corresponded to three predicted mentions before we applied this filter: ”der Landesvorsitzende Hans Taake” , ”Hans Taake”, and ”Taake”.

2. We removed the final word of a mention if its part of speech was tagged with any of the following: ”PREP”, ”KOKOM”, ”KON”, ”PTKNEG”, or ”ADV”*. This is because we observed that the algorithm otherwise would generate some erroneous mentions, such as ”Hans Taake als” (”Hans Taake as”).

3. We discarded any mentions where the headword was a pleonastic pronoun (”es” [”it”] in expressions like ”es regnet” [”it rains”]), a degree noun (e.g. ”Stunden” [”hours”]), a time noun (e.g. ”Nacht” [”night”]), or a generic pronoun (”man” [”you”/”one”]).

The pleonastic pronoun filter is based on a corresponding filter by Lee et al. [1]. However, the other filters were created on the basis of our own

*For descriptions of the tags, see the STTS guidelines [14].

observations during the experimental phase of this project. Some further filters are used by Lee et al. [1], but these are tailored to the English corpus used in their system and we did not find them relevant for our own system.

3.2.2 The sieve algorithm

The sieve algorithm takes as input the mentions. It then runs the mentions through a series of sieves in order to gradually cluster the mentions together. At the end of the algorithm, it outputs the final clusters. Here, we will explain the main steps of the sieve algorithm. The algorithm in our program corresponds to the concept of Lee et al. [1], however it was based on the pseudocode by Vällfors [63] because of the pseudocode's simple and easily comprehensible nature.

At the beginning of the algorithm, all mentions are made singletons, i.e. put in their own clusters. Then we iterate through all of the sieves. In our case, we used ten sieves which are explained further in section 3.2.3. The sieves are ordered such that high-precision* sieves are placed first. For each sieve, we iterate through the mentions relevant for coreference resolution. The relevant mentions in our program were chosen according to the mention selection process described in section 3.2.2.1. For each mention, we sort all of its antecedents, i.e. all mentions that precede the current mention in the text. Those are considered candidates for coreference. The sorting of antecedents in our program is described in section 3.2.2.2.

The next step is to iterate through the mention's antecedents. The current mention-antecedent pair is input to the current sieve function, and the sieve then decides whether or not the mentions corefer according to some given rules. A sieve may enable feature sharing as described in section 3.2.2.3. If the sieve returns true, the mention-antecedent pair is seen as a match, and their clusters are merged. If the sieve returns false, we continue the iteration of antecedents and repeat the process for the next mention-antecedent pair. Once a sieve has returned true for the current mention, or when all of its antecedents have been iterated through, the algorithm moves on to resolve the next mention. When all the relevant mentions have been considered by the current sieve, the whole process is repeated for the next sieve. This may result in some mentions still being alone in their own clusters at the end of the algorithm, if no sieve returns true for that mention.

When the algorithm has finished, it outputs the resulting clusters. The clusters can then be post-processed if necessary in order to match the gold

*The term "precision" refers to the term defined in section 2.2.5.

standard clusters. The resulting clusters of our algorithm were post-processed according to the description in section 3.2.2.4.

3.2.2.1 Mention selection

In the Stanford sieve algorithm, only the mentions that appear first in their clusters are considered, i.e. we only consider the mention in a cluster that appear furthest to the left in the text. This feature is motivated by, among others, the fact that mentions appearing early are less likely to have many modifiers or be pronouns [1].

In addition to this, we applied three filters based on the filters in Lee et al. [1]. The purpose of the filters was to remove mentions that may refer to general phenomena rather than specific entities. Thus, we discarded any mentions that fulfilled any of the criteria below.

- **Indefinite pronoun:** The mention contains an indefinite pronoun. We used the list of German indefinite pronouns in Andersson [8]. This list features single-worded modifiers, e.g. ”beide” (”both”), multiple-worded modifiers, e.g. ”ein bisschen” (”a little”), and heads, e.g. ”niemand” (”nobody”).
- **Indefinite article:** The mention starts with an indefinite article, e.g. ”ein” (”a”).
- **Bare plural:** The mention is a bare plural, i.e. a plural without any determiners or modifiers, e.g. ”Hunde” (”dogs”).

An exception was the Exact match sieve (see section 3.2.3.1), where only the Bare plural filter was applied, in accordance with Lee et al. [1]. For all of the other sieves, all three filters were applied.

3.2.2.2 Antecedent selection

In the Stanford sieve approach, it is important for the antecedents to be ordered by relevancy, such that the most relevant antecedents come first. This is because whenever a sieve returns true for a mention-antecedent pair, no further antecedents for that mentions are considered. This ordering of antecedents is called antecedent selection. The antecedent selection in Lee et al. [1] is based on syntax trees. As already established in section 3.2.1, we only had access to dependency trees. Therefore, for our dependency tree-based antecedent

selection, we again referred to Vällfors [63]. The antecedent selection of our program was based on her pseudocode.

For the current mention, we iterated through all of its antecedents. All antecedents appearing in another sentence than the mention were ordered by distance from the mention's sentence, shortest distance first and longest distance last. All antecedents appearing in the same sentence as the mention were ordered such that subjects were placed first*. In the final list, the antecedents from the same sentence as the mention were placed before the antecedents from other sentences.

3.2.2.3 Feature sharing

As mentioned in section 2.2.3.1, a key factor of the Stanford sieve approach is the entity-centric aspect. This enables features to be shared between all the mentions of a cluster. This means that whenever a mention-antecedent pair is input to a sieve, the sieve can consider the features not only of the current mention and antecedent, but also of the mentions appearing in their corresponding clusters [1].

In our algorithm, we enabled gender, number, and NER label for feature sharing. However, contrary to Lee et al. [1], we only enabled feature sharing in some of the sieves, not all. We disabled feature sharing where we thought that it could lead to erroneous matches, or where it did not make sense to enable it because of the way a sieve function was written. Gender and number sharing was disabled for the Exact match sieve, the Relative pronoun rule of the Precise constructs sieve, the Cluster head match rule of the Strict head match sieve, and the Relaxed head match sieve. Gender sharing was also disabled in the GermaNet sieve. NER label sharing was disabled in the NER sieve.

3.2.2.4 Post-processing

After the main algorithm had finished its clustering procedure, all singletons were removed, similar to Lee et al. [1]. This post-processing was needed in order to match the predicted clusters with the clusters of our gold standard, which only features clusters containing at least two mentions.

*For more information on subjects, we refer to the "nsubj" tag of Universal Dependencies: https://universaldependencies.org/u/dep/nsubj.html.

3.2.3 The sieves

Here, we present the ten sieves that were applied in our system. The sieves were mainly inspired by the sieves in the Stanford sieve approach and the sieves in CoRefGer-rule. As CoRefGer-rule appears to be based on the 2010 version of the Stanford sieve approach by Raghunathan et al. [5], the term ”Stanford sieve approach” in this section will refer to that version unless otherwise stated.

3.2.3.1 Sieve 1: Exact match

The purpose of this sieve is to link mentions only if they contain the same words. Our version is similar to the corresponding sieve in the Stanford sieve approach, but has been extended in accordance with CoRefGer-rule in order to adapt the sieve to the German grammar. Furthermore, we decided to disable this sieve for single pronouns, because we observed that those yielded errors in the output by e.g. putting all ”er” (”he”) pronouns of a text in the same cluster. This sieve returns true if any of the following two rules returns true:

- **Exact string match:** Returns true if the mention’s and antecedent’s strings are exactly equal. For example, the mention string ”der braune Hund” (”the brown dog”) only returns true for the antecedent string ”der braune Hund”.
- **Lemma match:** Returns true if the mention’s and antecedent’s words are exactly the same lemmas, with the same gender, number, and, if applicable, person, in the same order. This rule is needed for German in order to make it possible for noun phrases to match if their only difference is the case. For example, the rule returns true for ”der braune Hund” (”the brown dog”, nominative) and ”den braunen Hund” (”the brown dog”, accusative).

  This rule is not featured in the Stanford sieve approach as it is unique for German. It is featured in CoRefGer-rule, however as they did not have access to a morphological analyzer, they instead had to compare the words with their endings removed.

3.2.3.2 Sieve 2: Precise constructs

This sieve links mentions that have certain forms. The sieve is similar to the corresponding sieve in the Stanford sieve approach, but contains some adaptions to the German grammar. The sieve has also been implemented by

CoRefGer-rule. For this sieve to return true, any of the following six rules must return true:

- **Apposition:** Returns true if the mention is an apposition to the antecedent. This requires that the headword of the mention has been annotated with the apposition dependency relation, and that no other mention comes in between the mention and antecedent. For example, it returns true for ”[Hans Taake], [der Landesvorsitzende]” (”[Hans Taake], [the regional chairman]”).

- **Predicate nominative:** Returns true if the mention and antecedent are in a copulative relation, i.e. that one is a predicative nominative of the other. This requires that one of the headwords is annotated with the subject relation and that the other is annotated with the predicate relation. Furthermore, they must both be in the nominative case [8], have the same number, and no other mention can come in between them. For example, this rule returns true for ”[Er] ist [der Landesvorsitzende]” (”[He] is [the regional chairman]”).

- **Role appositive:** Returns true if the mention’s headword is a noun and the antecedent appears as a modifier in the mention. This also requires that the mention has the person NER label and that the antecedent headword has the GermaNet category label ”Mensch” (”human”). For example, the rule returns true for ”[[Schauspielerin] Marlene Dietrich]” (”[[actress] Marlene Dietrich]”).

  In the Stanford sieve approach, it is also required that the gender of the antecedent is not neutral. While noun references to persons in German in most cases are masculine or feminine, based on the natural gender of the person they refer to, some neuter references still exist. Therefore, we did not include this requirement.

- **Relative pronoun:** Returns true if the mention is a single relative pronoun referring to the antecedent’s headword. This requires that the gender and number of both mentions correspond, and that no other mention comes in between them. For example, this rule returns true for ”[der Hund], [der] im Park spielt” (”[the dog], [that] plays in the park”).

- **Abbreviation:** Returns true if the mention is an abbreviation of the antecedent. As previously stated, German abbreviations can take many forms, and it is not possible to cover all imaginable cases. Therefore,

we decided to only consider abbreviatons that consist of either all the initial capitalized letters, or all initial letters. This also takes hyphens into account. For example, this rule returns true for the mention ”CSU” and the antecedent ”Christlich-Soziale Union in Bayern”, and for the mention ”AfD” and the antecedent ”Alternative für Deutschland” (both examples being German political parties).

The Stanford sieve approach only considers abbreviations consisting of initial capitalized letters. However, we extended it to cover more cases in German.

- **Demonym:** Returns true if one of the mentions is a plural demonym of the other. For example, it returns true for ”die Schweiz” (”Switzerland”) and ”Schweizer” (”Swiss”). We used the country names and demonyms from EU's German list of countries, territories and currencies*. For simplicity reasons, we ruled out entries where the demonym had a deviant form (e.g. containing ”von” and ”der”), female demonym forms (as usually only the male form is used to denote a group), and some dual country names.

  The demonym list used for CoRefGer-rule is a translated version of the English Wikipedia list of demonyms used by Raghunathan et al. [5]. In our case, we thought it a better alternative to use an existing list in German.

3.2.3.3 Sieve 3: Strict head match

The purpose of this sieve is to link mentions based on their headword. However, it avoids the simple rule of linking all mentions with the same headword, as this can lead to errors when a mention's modifier is decisive. Our version is nearly identical to that of the Stanford sieve approach, with some small adaptions to the German grammar. The sieve returns true if all of the following four rules return true:

- **Cluster head match:** Returns true if the mention's headword is equal to any of the headwords in the antecedent cluster. In our case, this requires that the lemma, gender, and number of the headwords correspond.
- **Word inclusion:** Returns true if all the words in the mention cluster appear in the antecedent cluster. In our case, we only compared lemmas.

*https://publications.europa.eu/code/de/de-5000500.htm

The rule also excludes words that appear in a so-called stop word list, which a list of words bearing insignificant meaning, such as articles. We used the CoRefGer-rule stop word list*.

- **Compatible modifiers:** Returns true if all of the mention's modifiers appear as modifiers in the antecedent. This rule only looks at modifier words which are nouns or adjectives.
- **Not i-within-i:** Returns true if the mention is not a subset of the antecedent, or vice versa.

While Srivstava et al. [4] state that they implemented this sieve in CoRefGer-rule, they also state that they had problems implementing it. The details of their own implementation, however, are not described in their article. Therefore, we simply followed the Stanford sieve approach in the creation of this sieve.

3.2.3.4 Sieve 4: Strict head match variant 1

This sieve is identical to the corresponding sieves in the Stanford sieve approach and CoRefGer-rule. It is a less extensive variant of the Strict head match sieve. It returns true if all of the following rules of the Strict head match sieve return true: Cluster head match, Word inclusion, and Not i-within-i.

3.2.3.5 Sieve 5: Strict head match variant 2

Similar to the previous sieve, this is another less extensive variant of the Strict head match sieve, as used in both the Stanford sieve approach and CoRefGer-rule. It returns true if all of the following rules of the Strict head match sieve return true: Cluster head match, Compatible modifiers, and Not i-within-i.

3.2.3.6 Sieve 6: Relaxed head match

This sieve is similar to the corresponding one in the Stanford sieve approach. It has also been utilized by CoRefGer-rule. The sieve returns true if the mention's headword appears in the antecedent cluster. In our case, the words must correspond in lemma, gender, and number. Furthermore, a prerequisite is that both the mention and antecedent are named entities with the same NER label.

*https://github.com/dkt-projekt/e-NLP/blob/master/src/main/resources/stopwords/german.ser

3.2.3.7 Sieve 7: NER

This sieve is inspired by the NER sieve in CoRefGer-rule and based on a filter in Tuggener [41]. Its purpose is to match named entities. It returns true in any of the following two rules returns true:

- **Complete match:** Returns true if the mention and candidate contain the same named entity words in the same order that all have the same NER label. For example, it matches ”die Schauspielerin Marlene Dietrich” (”the actress Marlene Dietrich”) with ”Marlene Dietrich”, because they both contain the words ”Marlene Dietrich” with the person NER label.
- **Partial match:** Returns true if all of the mention’s named entity words are contained in the antecedent’s named entity, provided that both mention and antecedent have the same NER label. This also requires that the antecedent consists of more than a single word. For example, this rule returns true for the mention ”Dietrich” and the antecedent ”Marlene Dietrich”.

The Stanford sieve approach does not include a NER sieve similar to ours. However, the 2013 version of the algorithm [1] does include a sieve that performs proper head matching. We did not consider that sieve for reasons explained in section 3.2.3.11.

3.2.3.8 Sieve 8: Pronouns

This sieve is based partly on the pronoun sieve in the Stanford sieve approach and partly on the filters for German pronouns in Klenner, Fahrni, and Sennrich [39]. It’s purpose is to match a mention with an antecedent if the mention is a pronoun and the antecedent matches that pronoun on the basis of some given conditions. A prerequisite for this sieve is that there are at most three sentences between the mention’s sentence and antecedent’s sentence [1]. Furthermore, the following conditions apply:

- **3rd person plural pronouns:** Returns true if the mention is a 3rd person plural pronoun and the antecedent is also plural. Else, it returns false.
- **Possessive pronouns:** If the mention is a possessive pronoun and the antecedent is a pronoun, this rule returns true if they have the corresponding person and gender. If instead the antecedent is a noun,

the rule returns true if they have the corresponding gender [39]. Else, it returns false.

- **Reflexive pronouns:** If the mention is a reflexive pronoun, this rule returns true if the mention and antecedent are in the same clause [39]. Else, it returns false.
- **Personal pronouns:** If both the mention and antecedent are personal pronouns, this rule returns false if they are in the same clause [39].

For the mentions that are not covered by any of the cases above, the sieve returns true if the antecedent is a pronoun and corresponds with the mention in gender, number and person, or in gender and number otherwise.

Our pronoun sieve differs from the one in the Stanford sieve approach mainly because that approach also includes the use of animacy and NER labels, which our approach does not. Animacy and NER labels were considered not relevant for our sieve since pronouns of all genders can be used for animate as well as inanimate entities in German.

Due to lack of grammatical features, CoRefGer-rule did not have the possibility to implement a pronoun sieve. Thus, such as sieve was an improvement suggestion by Srivastava et al. [4].

3.2.3.9 Sieve 9: GermaNet

The purpose of this sieve is to add the semantic aspect of words. It is based on the semantic sieve in Krug et al. [7] and the semantic filters of Klenner, Fahrni, and Sennrich [39]. It returns true if one of the mentions' headwords is a synonym, hyperonym, or hyponym of the other mention's headword, and they have the same number. We also added a gender restriction for the synonymy relation: if both headwords are labeled with the "Mensch" ("human") category by GermaNet, and one of the genders is masculine while the other is feminine, the sieve returns false. This is to avoid that words such as "Lehrer" ("male teacher") and "Lehrerin" ("female teacher") are matched.

A corresponding sieve is not available in the Stanford sieve approach. However, adding a sieve based on GermaNet was an improvement suggestion by Srivastava et al. [4].

3.2.3.10 Sieve 10: Word vectors

This is an experimental sieve with no equivalent in the Stanford sieve approach or CoRefGer-rule. Like the previous sieve, it utilizes the semantic aspect.

First, all the words of the mention and antecedent are converted into fastText word vectors. This process excludes articles and pronouns as we judged that they have little semantic value. Then, for the mention and antecedent respectively, we calculate the mean vector based on all of their word vectors. Finally, we compare the mention and antecedent mean vectors by calculating the cosine similarity. If it exceeds 0.70, this sieve returns true. If a word vector for any word in the mention or antecedent (excluding articles and pronouns) are not available, this sieve simply returns false.

This sieve was included to add an additional semantic aspect to the algorithm. Furthermore, it does not only consider a mention's headword, but several (or all) of the words in its string. The cosine similarity value of >0.70 was chosen based on experimental assessment.

3.2.3.11 Omitted sieves from the original algorithm

There are three sieves featured in the latest (2013) version of the Stanford sieve algorithm [1] which we did not apply in our system: Speaker identification, Relaxed string match, and Proper head word match. The purpose of the Speaker identification sieve is to "[match] speakers to compatible pronouns" [1]. In other words, this sieve is relevant for quotations such as "'[I] feel happy today', [he] said", in order to determine that "I" corefers with "he". We did not implement this sieve because neither the parser nor the gold standard information that we used features speaker detection. This sieve is also not used in CoRefGer-rule, as it is based on the 2010 version of the Stanford sieve algorithm [5], where the Speaker identification sieve was not yet introduced.

The Relaxed string match and Proper head word match are different variants of head matching sieves. They were also introduced in the 2013 version of the Stanford sieve algorithm, and were not applied in our system because they were overlooked. As already mentioned, CoRefGer-rule is based on an older version, which is the one we mostly referred to when implementing our sieves. Thus, we simply missed to include these two new sieves. We will expand on this mistake in section 5.

Chapter 4

Results

4.1 Mention detection

As the gold mentions do not include singletons, a direct comparison between the predicted mentions and the gold mentions was not possible. This made it difficult to evaluate the mention detection part of our system. We therefore chose to evaluate the mention detection in two different ways: setting 1 and setting 2. The settings differ in which predicted mentions they include. Setting 1 includes all of the mentions predicted by our mention detection algorithm, i.e. singletons are included. Setting 2 only includes the predicted mentions in the resulting clusters that contain at least two mentions, i.e. singletons are excluded. As described in the next section, 4.2.1, we evaluated different versions of our sieve algorithm, including different sieves at a time. The version we used for evaluating mention detection in setting 2 was the one evaluated on predicted mentions that achieved the majority of the highest F1 scores. This version included the first eight sieves.

Table 4.1 shows the precision, recall, and F1 score for both settings respectively.

	P	R	F1
setting 1 (singletons included)	26.38	73.48	38.82
setting 2 (singletons excluded)	43.96	57.45	49.81

Table 4.1: Precision, recall, and F1 scores for the mention detection part of our model.

In both settings, recall was higher than precision. In setting 1, precision was 26.38, while recall was 73.48. In setting 2, the difference between the

scores was smaller than in setting 1: precision was 43.96 and recall was 57.45. Precision was the highest in setting 2, whereas recall was the highest in setting 1. The highest F1 score, 49.81, was achieved in setting 2. The scores averaged over both settings were 35.17 for precision, 65.47 for recall, and 44.32 for the F1 score.

Looking through a sample of all the mentions predicted by our algorithm (setting 1), some false positives, i.e. predicted mentions not among the gold mentions, were the following:

(1) die Friedhofsgärtner
(*the cemetery gardeners*)

(2) Politik
(*politics*)

Both examples constitute complete noun phrases. Example (1) is a mention that would have been reasonable to use in a coreference chain, although this was not the case in the article in question. Example (2) is more unlikely to be used for coreference resolution, as it can refer to a general phenomenon. However, it all depends on the context, and therefore we cannot for sure say that this mention would never be relevant.

Looking at some false negatives, i.e. gold mentions not found by our mention detection algorithm, we observed several cases where the gold mention was not ignored, but only part of it was found. For example, the gold mention in example (3) corresponded to three different predicted mentions, and only a part of the gold mention in example (4) was found.

(3) **Gold mention:**
die PionierInnen der ”Hamburger Ehe”, drei lesbische und vier schwule Paare
(*the pioneers of the ”Hamburg Marriage”, three lesbian and four gay couples*)
Predicted mentions:
die PionierInnen – Hamburger Ehe – vier schwule Paare
(*the pioneers – Hamburg Marriage – four gay couples*)

(4) **Gold mention:**
die bald nicht mehr städtischen HEW
(*the soon-to-be no longer urban HEW [Hamburgische Electricitäts-Werke]*)

Predicted mention:
städtischen HEW
(*the urban HEW [Hamburgische Electricitäts-Werke]*)

4.2 Coreference resolution

In this section, we will present the precision, recall and F1 scores achieved by our coreference resolution algorithm. After that, we will expand on the contribution of individual parts of the algorithm. Lastly, a comparison of our model and other German coreference resolution models will be made.

4.2.1 Ablation studies

Our sieve algorithm was evaluated with some sieve added successively, in order to determine the contribution of individual sieves. This evaluation method is inspired by the one in Lee et al. [1]. However, not all sieves were added successively. We chose to only perform ablation studies on sieves that were more or less unique to our system in comparison with the 2010 version of the Stanford sieve algorithm [5]. These include the last four sieves: NER, Pronouns, GermaNet, and Word vectors. This evaluation method also required a base system to which the sieves could be successively added. Naturally, our base system consisted of the first six sieves.

The evaluation of the sieve algorithm with sieves added successively was performed twice. First, it was performed using the mentions predicted by our mention detection algorithm, and second, it was performed using the gold mentions. Evaluating the system with both predicted and gold mentions is common practice within coreference resolution research [1]. While predicting mentions is an important part of the coreference resolution pipeline in real-life settings, the use of gold mentions enables evaluation of the resolution algorithm alone. The MUC, B^3, CEAF, and CoNLL metric scores for our coreference resolution system are presented in tables 4.2 and 4.3 below.

The best performance was achieved using gold mentions. The system version that included all of the sieves achieved the highest MUC, B^3, and CoNLL F1 scores overall: 69.81, 57.68, and 63.50 respectively. The highest CEAF F1 score was achieved by the version that included all of the sieves except the Word vectors sieve: 63.51. For predicted mentions, the highest-scoring system version in general was the one that included the NER and Pronouns sieves, achieving a CoNLL F1 score of 31.16. This corresponds to approximately half of the highest CoNLL F1 score using gold mentions.

	MUC			B^3			CEAF			CoNLL
	P	R	F1	P	R	F1	P	R	F1	F1
base	31.33	21.57	25.55	25.38	18.55	21.43	27.88	30.08	28.93	25.31
+NER	**31.52**	22.30	26.12	25.65	19.22	21.98	**28.32**	30.62	29.42	25.84
+Pronouns	27.04	31.86	**29.26**	**38.34**	28.79	32.88	24.76	**42.67**	**31.34**	**31.16**
+GermaNet	25.71	32.45	28.69	38.13	29.44	33.23	23.77	42.56	30.51	30.81
+Word vectors	24.99	**32.98**	28.43	37.93	**29.96**	**33.48**	23.43	42.57	30.22	30.71

Table 4.2: Precision, recall, and F1 scores for our model using predicted mentions.

	MUC			B^3			CEAF			CoNLL
	P	R	F1	P	R	F1	P	R	F1	F1
base	87.26	40.53	55.35	43.36	34.60	38.49	74.16	43.43	54.78	49.54
+NER	**87.29**	40.71	55.52	43.41	34.83	38.65	**74.46**	43.46	54.88	49.68
+Pronouns	80.79	60.19	68.98	62.28	51.80	56.56	68.72	**58.98**	63.48	63.01
+GermaNet	80.47	61.33	69.61	**62.44**	52.90	57.28	68.92	58.90	**63.51**	63.47
+Word vectors	79.98	**61.93**	**69.81**	61.88	**53.67**	**57.48**	68.89	58.37	63.19	**63.50**

Table 4.3: Precision, recall, and F1 scores for our model using gold mentions.

The sieve that yielded the biggest increase in CoNLL F1 score points compared to the previous system version was, in the case of both predicted and gold mentions, the Pronouns sieve. For predicted mentions, the increase was 5.32 points, and for gold mentions, the increase was 13.33 points. Adding the GermaNet and Word vectors sieves, however, lead to some decreased F1 scores.

In the case of precision and recall, precision was higher in earlier versions, and recall was higher in later versions, for both predicted and gold mentions. In general, precision decreased with the addition of sieves while recall increased, although there was variation in this aspect. The highest precision, 87.29, and the highest recall, 61.93, were achieved using gold mentions and the MUC metric.

In the following, we will expand on the contribution of each part of the system: the base system, the NER sieve, the Pronouns sieve, the GermaNet sieve, and the Word vectors sieve.

4.2.1.1 Contribution of the base system

Using only the first six sieves, the system yielded a CoNLL F1 score of 25.31 for predicted mentions and 49.54 for gold mentions. In other words, the use of predicted mentions lead to achieving about half of the score achieved with the use of gold mentions.

4.2.1.2 Contribution of the NER sieve

The addition of the NER sieve provided higher F1 scores for all metrics compared to the base system, in the case of predicted as well as gold mentions. However, the exact increases were low for both settings. For predicted mentions, the CoNLL F1 score went from 25.31 to 25.84, yielding a score increase of 0.52 points. For gold mentions, the CoNLL F1 score went from 49.54 to 49.68, yielding a score increase of 0.14 points.

Looking through a sample of the mention-antecedent pair matches yielded by this sieve, we observed some correct matches, such as:

(5) Coles – Phil Coles

(6) das PTU – Institut für Polizeitechnische Untersuchung (PTU) des Landeskriminalamtes (LKA)
(*the PTU – Institute for Police Technology Investigations [PTU] of the State Criminal Police [LKA]*)

However, we also observed some erroneous matches, such as:

(7) Helena – Helena und Cecilia Bergman
(*Helena – Helena and Cecilia Bergman*)

(8) Bonn – der rot-grünen Bonner Koaliton
(*Bonn – the red-green coalition of Bonn*)

In example (7), the match is erroneous because the first mention refers to a single person, whereas the second mention refers to two people (the single person being one of them). In example (8), the first mention refers to a city, and the second mention refers to a political organization within that city, which is also an incorrect match.

4.2.1.3 Contribution of the Pronouns sieve

Adding the Pronoun sieve increased the F1 scores for both predicted and gold mentions. In particular, looking only at the predicted mentions, this version of the system yielded the highest F1 scores for all metrics except B^3. Compared to the previous system, the CoNLL F1 score for predicted mentions went from 25.84 to 31.16, yielding an increase of 5.32 points. The CoNLL F1 score for gold mentions increased by even more points, going from 49.68 to 63.01, yielding an increase of 13.33 points. An interesting observation is that while all of the F1 scores for gold mentions increased relatively steadily, the F1 score increases for predicted mentions varied notably. The MUC F1 and CEAF F1

score increases were only 3.14 and 1.92 points respectively, whereas the B^3 F1 score increase was 10.90 points.

Looking through a sample of the mention-antecedent pair matches yielded by this sieve, we observed several correct matches, such as:

(9) ich – mich
(*I – me*)

(10) er – der Polit-Kabarettist Jürgen Timm
(*he – the political cabaret artist Jürgen Timm*)

However, we also observed some erroneous matches, such as:

(11) seiner – seiner Firma
(*his – his company*)

(12) unser – unser Stadt
(*our – our city*)

Both are examples of mention-pairs with non-matching features. In example (11), a possessive pronoun referring to a masculine owner is matched with a feminine noun, which is incorrect because they have different genders. In example (12), a possessive pronoun referring to a plural owner is matched with a singular noun, which is incorrect because they have different numbers.

4.2.1.4 Contribution of the GermaNet sieve

The GermaNet sieve addition had a small impact on the F1 scores for predicted as well as gold mentions. While the addition of the sieve yielded a CoNLL F1 score increase for gold mentions, it yielded a decrease for predicted mentions. The CoNLL F1 score for gold mentions went from 63.01 to 63.47, increasing with 0.46 points, whereas the corresponding score for predicted mentions went from 31.16 to 30.81, decreasing by 0.35 points. However, this version of the system achieved the highest CEAF F1 score for gold mentions.

When we looked through the mention-antecedent pair matches yielded by this sieve, we observed that some matches using predicted mentions involved mentions irrelevant for coreference resolution, as in the examples (13) and (14) below.

(13) 22.30 Uhr – der Zeit
(*22.30 – the time*)

(14) ersten Moment – Zeitpunkt
(*the first moment – point in time*)

We also observed several erroneous matches where the mentions' modifiers were decisive, such as:

(15) das irakische Regime – weder die türkische Regierung*
(*the Iraqi regime – neither the Turkish government*)

(16) die SPD-Abgeordnete Erika Woisin – die Abgeordnete Bettina Macha-czek
(*the SPD representative Erika Woisin – the representative Bettina Macha-czek*)

In both examples, the headwords are in a hyperonomy/hyponomy relationship. In example (15), the headword "Regime" ("regime") is a hyponym of the headword "Regierung" ("government"). However, looking at the mentions' modifiers, we can see that they refer to different entities: Iraq and Türkiye, therefore this match is erroneous. Similarly, in example (16), "SPD-Abgeordnete" ("SPD representative") is a hyponym to "Abgeordnete" ("representative"). However, the modifiers "Erika Woisin" and "Bettina Macha-czek", which are names of people, make it clear that the mentions refer to different entities.

Nonetheless, this sieve still yielded some correct matches, as in the examples below.

(17) der Allianz – der westlichen Militärallianz
(*the alliance – the Western military alliance*)

(18) die Hymne – das Lied eines gewissen "Tony"
(*the anthem – the song of a certain "Tony"*)

4.2.1.5 Contribution of the Word vectors sieve

Similar to the addition of the GermaNet sieve, the addition of the Word vectors sieve had a small impact on the F1 scores for both predicted and gold mentions. Again, it yielded a CoNLL F1 score decrease for predicted mentions and a CoNLL F1 score increase for gold mentions. However, the score differences

*The predicted mention "weder die türkische Regierung" in itself is erroneous, as a noun phrase cannot start with "weder" ("neither") in this way. However, the word's existence does not influence the outcome of this sieve as the sieve only considers headwords, and thus this mention can still be used to prove the point.

compared to the previous version were even smaller. The score for predicted mentions decreased by 0.10 points, from 30.81 to 30.71, while the score for gold mentions increased by 0.03 points, from 63.47 to 63.50. This version of the system provided the highest F1 score for gold mentions for all metrics except CEAF, and provided the highest CoNLL F1 score overall.

When looking through a sample of matches made by this sieve, we noticed that a huge share of the mentions included proper names. Some of the correct matches were:

(19) Umweltminister Jürgen Trittin – Trittin
(*environment minister Jürgen Trittin – Trittin*)

(20) Regisseur McDougall – Charles McDougalls
(*director McDougall – Charles McDougalls*)

We also observed several mentions with close but not identical meanings, leading to erroneous matches as in the following examples.

(21) Olaf Ludwig – Uwe Ampler

(22) Neukölln – Kreuzberg

”Olaf Ludwig” and ”Uwe Ampler” in example (21) are both names of East German cyclists. In example (22), ”Neukölln” and ”Kreuzberg” are both names of districts in Berlin. In these examples, the entities are semantically close but not the same, which makes them erroneous matches.

Other errors occurred when a single word or a few words in a mention seemed to dominate the final value of its corresponding word vector:

(23) Benno Ohnesorg und Rudi Dutschke – Rudi Dutschke
(*Benno Ohnesorg and Rudi Dutschke – Rudi Dutschke*)

(24) neue Klage gegen VW von NS-Opfern – VW
(*new lawsuit against VW [Volkswagen] by victims of the National Socialist regime – VW [Volkswagen]*)

Similar to example (7), one of the mentions in example (23) denotes a single person whereas the other mention denotes two people, the single person being one of them. In example (24), the only common factor is the word ”VW” (”VW [Volkswagen]”). While looking at the mentions as a whole, they clearly refer to different entities (a lawsuit versus a company).

4.2.2 Comparison to previous work

Table 4.4 shows a summary of the F1 scores achieved by some coreference resolution models evaluated on German news texts, including our own model. We excluded models that were only evaluated on literature corpora, as coreference resolution within the field of literature often has a narrower scope, focusing mainly on the identification of characters. The numbers in the table refer to the highest scores achieved for each model. In most of the cases, gold mentions were used, however this information is not available for all models' evaluations. Furthermore, not all evaluations of previous models reported scores for all of the metrics used in our own evaluation, e.g. some of them only reported CoNLL F1 scores. For one of the models, an F1 score was reported, but the exact metrics used to calculate it were not specified. Therefore, this score was placed in an "other" column.

system	corpus	MUC	B^3	CEAF	CoNLL	other
this work*	TüBa-D/Z	69.81	57.48	63.51	63.50	–
Klenner et al. [39]†	TüBa-D/Z	–	–	–	–	61.49‡
CorZu [38]§	SemEval-2010	–	–	–	58.11	–
IMS HotCoref DE [44]¶	TüBa-D/Z	–	–	–	65.76	–
	SemEval-2010	–	–	–	63.61	–
CoRefGer-rule [4]‖	TüBa-D/Z	70.50	23.10	–	–	–
	SemEval-2010	50.20	63.30	–	–	–
Schröder et al. [3]**	TüBa-D/Z	**82.23**	**77.05**	**77.09**	**78.79**	–
	SemEval-2010	77.77	72.14	73.47	74.46	–

Table 4.4: Summary of F1 scores of different coreference resolution models evaluated on German news texts.

As we can see, the model of Schröder, Hatzel, and Biemann [3] evaluated on TüBa-D/Z achieved the highest F1 scores for all metrics, providing the current state-of-the-art performance. The difference between their achieved CoNLL F1 score, 78.79, and ours, 63.50, is 15.29 points. Out of the four models

*The scores refer to the use of gold mentions.

†The scores refer to the use of gold standard information.

‡The score was calculated using 5-fold cross validation. The exact metrics are not specified.

§The score refers to the use of gold mentions. The score was calculated by Rösiger and Kuhn [44], as Tuggener [38] only provides scores for anaphora resolution.

¶The TüBa-D/Z score refers to the use of gold mentions. The SemEval-2010 score refers to the use of gold mentions and the settings "open" and "regular" of the SemEval-2010 task.

‖The TüBa-D/Z scores refer to the use of gold mentions. Whether or not the SemEval-2010 scores refer to the use of gold mentions is not specified.

**The scores refer to the use of the coarse-to-fine large model, singletons excluded.

reporting CoNLL F1 scores, we outperformed CorZu with 5.39 points. In turn, IMS HotCoref DE outperformed our model with 2.26 points (evaluated on TüBa-D/Z) and 0.11 points (evaluated on SemEval-2010) respectively. Although a direct comparison to Klenner et al. [39] cannot be made due to their evaluation metrics being unknown, we can observe that our CoNLL F1 score is slightly higher than their reported F1 score, with a difference of 2.01 points.

CoRefGer-rule was evaluated using only MUC and B^3. Comparing our scores to their TüBa-D/Z corpus scores, we achieved a lower MUC F1 score, 69.81 as opposed to 70.50, but a higher B^3 F1 score, 57.48 as opposed to 23.10. If we compare our scores to their SemEval-2010 scores, the results are reversed: we achieved a higher MUC F1 score, 69.81 as opposed to 50.20, but a lower B^3 F1 score, 57.68 as opposed to 63.30.

Chapter 5

Discussion

5.1 General performance

Based on the summary of scores reported for German coreference resolution models, as shown in table 4.4, we can conclude that we did not achieve state-of-the-art performance. We did also not expect our rule-based model to outperform a deep learning model. Looking at the CoNLL and ”other” F1 scores, excluding the deep learning model, it appears that our model performs on par with the machine learning models. The difference between our score and the scores of these models are 2.01 (Klenner et al. [39]), 5.39 (CorZu [38]), 2.26 (IMS HotCoref DE [44]), and 0.11 (IMS HotCoref DE [44]). However, it is difficult to compare models based only on F1 scores. This is because the numbers do not say anything about the details of each model, such as strengths and weaknesses. Likewise, the evaluation metrics all have biases, and none of them can be seen as the one ”true” representation of a coreference resolution model’s performance.

5.2 Contribution of morphological analyzer

The morphological analyzer, ParZu, enabled the access to grammatical information about words. Grammatical information, such as a word’s lemma, part of speech, gender, case, number, person and definiteness, was utilized in all of our sieves. In other words, it was a fundamental part of our system. Since German is a highly inflectional language, where words change form based on grammatical circumstances, the access to grammatical information allowed us to make precise comparisons between mentions. That is, we did not have to modify or deconstruct words in order to extract information and

enable comparisons. Instead, we could utilize the grammatical information directly, which enabled precise and effective rules. For the Pronouns sieve in particular, the use of grammatical information was essential. Below, we will expand on the strengths and weaknesses of this sieve.

5.2.1 The Pronouns sieve

As a result of this sieve, we were able to match pronouns with nouns because we could compare their gender and number. Otherwise, the matching of such mentions would have been difficult, as pronouns' and nouns' forms do not correspond (e.g. ”Junge” [”boy”] bears no similarity with ”er” [”he”]). Furthermore, we could match pronouns with pronouns without them needing to have an identical or similar form. Matching pronouns based on their form can also lead to errors. For example, the possessive pronoun ”sein” (”his”/”its”) is used for both masculine and neuter entities. Furthermore, the pronoun ”sie” is used for 3rd person singular (”she”), 3rd person plural (”they”), and, if the initial letter is capitalized, a formal version of ”you”.

The Pronouns sieve achieved the greatest increase in F1 score for both predicted and gold mentions, and can thus be seen as our most successful sieve, out of the individual sieves compared in section 4.2.1. Since most of the sieves in our algorithm focused on the resolution of nouns, a pronoun-resolving sieve was a necessary and important part of our algorithm. Without the use of it, many pronouns are assumed to have been left unsolved.

However, despite its seemingly good performance, the Pronouns sieve was not without errors. The main error identified concerns possessive pronouns. A characteristic unique to possessive pronouns is that they actually have two genders and numbers: the gender and number of the owner entity, and the gender and number of the owned entity. In a possessive pronoun, the root morpheme is inflected according to the owner's gender and number, and the affix is inflected according to the owned's gender and number. For example, in the expression ”seine Katzen” (”his cats”), the possessive pronoun is ”seine”. The owner is masculine and singular, as shown in the base morpheme ”sein”, and the owned is feminine and plural, as shown in the affix ”-e”.

The dual gender and number features caused a problem in our case, because both ParZu and TüBa-D/Z only provides a single gender and number label for each word. When we looked through a sample of possessive pronouns, there seemed to be variation in whether the labels referred to the owner or the owned. In order to match a possessive pronoun with another mention, we needed the grammatical features of the owner. However, if the

grammatical features instead referred to the owned, this could cause erroneous matches, such as in examples (11) and (12).

The problem of the dual features was overlooked in the creation of our program. It could possibly be solved by analyzing the ParZu and TüBa-D/Z labelling in order to identify possible consistencies in how possessive pronouns are gendered and numbered.

5.3 Contribution of semantic information

Semantic information was incorporated into our system in two ways: through the GermaNet sieve and the Word vectors sieve. In both cases, the impact on the resulting F1 scores was minimal. For gold mentions, the increase of the CoNLL F1 score was only 0.46 points for the GermaNet sieve, and as little as 0.03 points for the Word vectors sieve. For the predicted mentions, the score even decreased, however still with small numbers. The decrease was 0.35 points and 0.10 points respectively. Looking at these numbers, the sieves can be seen as a negligible part of our system.

Lee et al. [17] conclude that one of the main error types in the Stanford sieve algorithm is the matching of a common noun with a common noun. They suggest that a reason is that semantics is required to resolve nouns that do not have the same form, e.g. ”victim” and ”casualty”. While it seems that our semantic sieves solved this problem to some extent, a new problem appeared: considering the semantics of more than a single or a few words. As we will see from the analysis of the GermaNet sieve and the Word vectors sieve below, both sieves yielded errors related to this problem.

In the next two sections, we will expand on the weaknesses of each of the sieves and analyze the reasons for their poor performance.

5.3.1 The GermaNet sieve

Regarding the GermaNet sieve, its purpose was to match synonyms and hyperonyms/hyponyms. While it did yield some correct matches, a great weakness of this sieve was that it only considered the headwords. This aspect led to the erroneous matches shown in examples (15) and (16). In both examples, it was made clear from the mentions' modifiers that they referred to different entities. Thus, the implementation of this sieve was possibly too simple.

Our GermaNet sieve was inspired by but not identical to the semantic sieve in Krug et al. [7] and the semantic filter in Klenner, Fahrni, and Sennrich

[39]. As Krug et al. [7] performed coreference resolution within the literature domain, they mainly focused on resolving novel characters. Thus, their semantic sieve only considered nouns referring to humans, which also required an agreement in gender. Furthermore, they only considered synonymy relations. This was considered a too narrow sieve for our project. Therefore, we also took inspiration from the semantic filter in Klenner, Fahrni, and Sennrich [39]. They considered both synonymy and hyperonomy/hyponomy relations and did not seem to exclude any semantic categories, as was the case with our sieve. However, they also filtered the antecedents by animacy and certain verb-based constraints. We did not involve animacy because we did not think it was relevant in our case. We also did not include verb-based constraints because that would require a complex sentence analysis, which we thought was outside the scope of our project.

An improvement suggestion for our semantic sieve could be to only consider nouns that denote humans, as in Krug et al. [7]. Although such a sieve would consider a narrower span of mentions, implementing it would probably solve the problem presented through examples (13) and (14). In these examples, predicted mentions not relevant for coreference resolution – time indications – are resolved. Although we removed mentions where the headword was a time or degree noun in the last step of the mention detection algorithm, this required the headword to be labeled with such information from the parsing process. Else, which was the case with the example mentions, the mentions were not filtered out. Another improvement suggestion for this sieve could be to implement the more advanced semantic filter in Klenner, Fahrni, and Sennrich [39].

However, although the suggestions above could possibly improve the GermaNet sieve, none of them solves the main problem identified at the beginning of this section: mentions where the modifiers are decisive. A suggestion could be to add a comparison of the mentions' modifiers, and add some filters based on possible cases. For example, if both mentions contain modifiers that are named entities, the sieve could compare these named entities and return false if they do not match, solving the error in example (16). Solving the error in example (15) is trickier: it would require an analysis of the adjectives.

To summarize, while our GermaNet sieve did not have much impact on the final result, some of the improvement suggestions could still be worth to try it in order to make the sieve more relevant. Nevertheless, the decisive modifier problem would be very difficult to solve completely, as it would require rules for all cases imaginable.

5.3.2 The Word vectors sieve

This experimental sieve had even less impact than the GermaNet sieve. While GermaNet only considered a mention's headword, the Word vectors sieve considered all of the mentions' words, except for articles and pronouns. In other words, we tried to cover a semantic aspect that was not covered by GermaNet. Although resulting in some correct matches, this sieve obviously had several weaknesses, providing the reasons for the poor results.

One of the main weaknesses, observed in examples (21) and (22), was that matches were made based on similarity and not equality. Mention-antecedent pairs only had to be "similar enough", i.e. exceed the arbitrarily chosen cosine similarity threshold of 0.70, to match. The mentions in these examples are semantically similar, yet not the same, which makes the matches erroneous in the case of coreference resolution. An increase in the cosine similarity threshold could possibly lead to a better result, as the matches have to be even more similar to match. However, this does still not guarantee the equality of the mention-antecedent pair, as it still only requires the pair to be "similar enough". Furthermore, in the experimental phase of our program development, increasing the cosine similarity with 0.1 points and above even worsened the F1 scores.

Another weakness was observed in examples (23) and (24). Here, it seemed as if one or several words in a mention dominated the mention's resulting mean word vector. A consequence was that some mentions were matched based on that word or those words independently of the other words in the mentions, which led to erroneous matches. This means that the aspect we focused on with this sieve – considering all of the mention's words – was not fully covered. The sieve still suffered from a problem similar to the main weakness of GermaNet.

All in all, our judgment is that the use of word vectors is not suitable for rule-based systems, because it leads to too many imprecise matches. We can also not see any improvement suggestions that would lead to the guarantee of equality rather than similarity matches, other than setting the cosine similarity threshold to 1.0. This would most likely only lead to matches where the mentions consist of the exact same words. However, as comparing the words of mentions is a simple task, there are more efficient ways to do it than using word vectors.

It is worth noting that we only tried out a single set of word vectors (fastText) and a single dimension (300). It is possible that using another set of word vectors and/or another dimension would have yielded better F1

scores. However, this fact does not change our concluding assessment about word vectors not being suitable for use, because the nature of the word vectors themselves will remain imprecise.

5.4 Comparison to CoRefGer-rule

Based on the F1 scores presented in table 4.4, it is difficult to determine whether or not our system outperformed CoRefGer-rule. Srivastava et al. [4] only reported scores for MUC and B^3, which means that we cannot compare CoNLL F1 scores. Furthermore, they received quite different results when evaluating on TüBa-D/Z and when evaluating on SemEval-2010. For MUC, the difference between their F1 scores was 20.30 points, and for B^3, the difference was 40.20 points. Our MUC score was higher than theirs for SemEval-2010, and our B^3 score was higher for TüBa-D/Z. As we cannot identify a clear pattern, we refrain from concluding anything based on these scores.

The fact that we implemented an independent system inspired by CoRefGer-rule and not used the CorRefGer-rule source code itself also complicates the comparison. Furthermore, it was not possible to create our own ”clone” of CoRefGer-rule as some of the sieves in CoRefGer-rule are not explained in detail in Srivastava et al. [4]. For example, regarding the Precise constructs sieve, they state that ”Due to the different tree tags a direct application of Stanford NLP algorithms was not possible” [4]. Yet, they do not explain their own implementation. Details of the matching criteria for their NER sieve are also not available. In these cases, we turned to the Stanford sieve algorithm [5] [1] and other sources for our own implementation.

Our system includes all of the sieves used in CoRefGer-rule and three additional sieves: the Pronouns sieve, the GermaNet sieve, and the Word vectors sieve. The additional sieves were added based on the improvement suggestions by Srivastava et al. [4], which were to add a morphological analyzer and a semantic sieve.

As mentioned previously, the morphological analyzer enabled the use of grammatical information. The incorporation of grammatical information was an important factor in our implementation of the sieves. While Srivastava et al. [4] accounted for variations in word endings in order to match words where their only difference was the case, we could compare lemma, gender, and number. We believe that our approach is more precise and efficient, as it enables direct comparisons between words without any modifications needed. Furthermore, grammatical information was useful throughout all of our sieves.

It also enabled the implementation of the Pronouns sieve. While Srivastava et al. [4] could only resolve pronouns that were an exact match, our Pronouns sieve provided a greater span of possibilities. It made it possible to match pronouns with nouns, and also to match pronouns of different types and cases.

Nonetheless, the GermaNet and Word vector sieves only had a small impact on our system and thus did not seem to be important additions. The suggestion of Srivastava et al. [4] was to add the semantic sieve used in Krug et al. [7]. We expanded this idea, leading to two different semantic sieves. However, none of them yielded a significant improvement.

5.5 Methodological weaknesses

Here, we will reflect on some weaknesses and sources of error of this project that were not already covered in the previous sections. However, a detailed error analysis will not be performed for the first six sieves. We considered this to be outside the scope, as our focus was on evaluating the sieves more unique to our system. An overview of the contribution of individual sieves in the original Stanford sieve algorithm is available in Lee et al. [1].

5.5.1 Parsing and mention detection errors

According to Lee et al. [17], the main source of error in their Stanford sieve model is the mention detection. In the evaluation of our own algorithm, the CoNLL F1 scores using predicted mentions corresponded to about half of the CoNLL F1 scores using gold mentions. In other words, the use of predicted mentions clearly caused a decline in the performance, which in turn is a consequence of imperfect mention detection.

Looking at table 4.1, which shows our mention detection scores, we see that the recall for setting 1 (singletons included) is high, 73.48. This means that a majority, but not all, of the gold mentions were found. We can also see that the precision was low, 26.38. This means that a majority of the predicted mentions were false positives, i.e. mentions that for some reason should not be used in any coreference resolution chains. From these observations, we identify the following two sources of error: first, not all mentions relevant for coreference resolution were found, and second, mentions not intended for coreference resolution risked being resolved by the sieve algorithm.

In order to understand why not all gold mentions were found, we have to analyze the mention detection algorithm. The mention detection algorithm forms mentions based on dependency relations. When a noun or pronoun has

been found, the algorithm searches left and right for dependency relations. In order for a token to be added to a mention, there has to be a relation from that token to any of the tokens already included in the mention. If we reach a token without a relevant dependency relation, this token is not added to the mention, and the search in that direction (left or right) is terminated. This means that the complete span of a mention will not be found if, for some reason, some of the tokens in the mention does not have a relation to any of the tokens already included.

The complete span of gold mentions not being found, as in the examples (3) and (4), seemed to be a common problem with our mention detection algorithm. One of the reasons was that punctuation marks, such as quotation marks, were provided with a dependency relation label that did not point to another token. This means that no mention could contain a punctuation mark, as reaching a punctuation mark would terminate the search. This problem could be solved by adapting the algorithm. Another reason was erroneous dependency relation labels produced by the parser. This could be a tricky problem to solve, because even if we would switch to another parser, perfect parsing is highly unlikely.

Moreover, erroneous parsing labels are not an isolated problem in the mention detection step. Erroneous mentions resulting from erroneous dependency relation labels risk leading to erroneous clustering. Furthermore, any errors in the labeling – wrong part of speech, wrong gender, wrong NER span, wrong NER label, etc. – would lead to errors in the sieves. In German, a factor that could cause erroneous labeling is the morphological ambiguity of pronouns and articles. In both cases, words exist that have the same form but different meanings. One example is the ambiguous ”sein” and ”sie”/”Sie” pronouns mentioned previously. Another example is the masculine and neuter articles, which both have the same dative and genitive forms. Klenner et al. [39] conclude that a source of error in their model was indeed ”the morphological ambiguity introduced by replacing perfect morphological descriptions with the output of a real morphological analyzer”. However, in their case, parsing errors yielded only a small drop in performance, but the extent of the problem is presumably parser dependent.

A further problem connected to the mention detection step was the fact that some mentions were correctly identified by our mention detection algorithm, but not relevant for coreference resolution in the current articles, as in examples (1) and (2). This is one of the reasons for the low precision in setting 1. As specified above, such mentions risked being resolved by our sieve algorithm, yielding false positives. This was the case in examples (13) and

(14). If we compare the precision scores for our model presented in tables 4.2 and 4.3, precision is, generally, notably lower for predicted mentions than gold mentions. Since precision was always less than 0.50 for predicted mentions, it means that a majority of the matches made were false positives – which is a result of erroneous or irrelevant mentions. The filtering of irrelevant mentions at the end of the mention detection step could be extended in order to reduce the error rate. However, we think that it is hard to get rid of this problem completely.

5.5.2 Simplified clause detection

Clauses were used in the Pronouns sieve in order to apply rules for reflexive and personal pronouns. However, neither ParZu nor TüBa-D/Z provides clause identification. Therefore, we had to construct our own clause detection. Since we considered it outside the project's scope to perform complex sentence analyzes, our clause detection relied on simplified assumptions about clauses. We assumed that all the words connected to the same root belong to the same clause. For predicted mentions, we travelled the dependency relations from the headword until we found a root, which identified its clause. For gold mentions, the same approach was difficult to apply because of the way we had already implemented the extraction of gold mentions. Thus, for gold mentions, we assumed that the majority of noun phrases are objects. This means that a mention's headword should be directly connected to the root. Therefore, the clause of a gold mention was simply identified by its headword's headword.

The weaknesses of our clause detection are that it was based on simplified assumptions, and that different detection approaches were used for predicted and gold mentions. The reason for using different approaches is that we regarded the first approach as more correct, and wanted to use it where possible. Furthermore, the second approach would not have been suitable for predicted mentions because of inconsistent dependency relations provided by the parser. In particular, this applied to multi-word verbs. If a word had a dependency relation to such a verb, there was inconsistency in which of the verb's words was identified as the headword.

5.5.3 NER sieve errors

The NER sieve yielded only a small improvement of the system performance: the CoNLL F1 score increased by 0.52 points for predicted mentions and by 0.14 points for gold mentions. While we do think that a NER sieve is a

relevant feature in a sieve algorithm, we admit that our implementation was not optimal. Looking at the erroneous examples (7) and (8), it appears that this sieve can be too inclusive. For the Partial match rule, it is only required that the mention's named entity is contained in the antecedent's named entity. Apparently, this led to errors, because the rule does not take into account special cases, such as ”und” (”and”) relations and words being part of other words. An improvement suggestion could be to add filters considering these aspects. Another possibility is to remove the Partial match rule completely, although the sieve then risks becoming too exclusive.

5.5.4 Consequences of omitted sieves

The omitted sieves from the Stanford sieve algorithm [1] were Speaker identification, Relaxed string match, and Proper head word match. The lack of the Speaker identification sieve meant that 1st person pronouns could only be resolved in our Pronouns sieve. In that sieve, we provided no specific rules for 1st person pronouns. According to our observations of a sample of all resulting matches made by our Pronouns sieve, the 1st person pronouns were matched with mentions that also contained 1st person pronouns. This approach led to a few correct matches, but was not a comprehensive solution. The Speaker identification sieve alone yielded a CoNLL F1 score of 29.20 in the Stanford sieve algorithm [1]. In conclusion, we assume that the addition of this sieve would have provided a non-negligible improvement to our system.

The purpose of the Relaxed string match sieve is to match mentions that, after removing all of the words following the headwords, are identical. It would have been an interesting sieve to add to our system. However, this sieve alone yielded an improvement of only 0.10 CoNLL F1 score points in the Stanford sieve model [1], which suggests that the lack of the Relaxed string match sieve in our system was probably not a significant oversight.

The Proper head word match sieve matches mentions that have the same proper noun headword. They must also meet three further requirements. First, one of the mentions must not be a subset of the other. Second, the modifiers cannot consist of ”different location named entities, other proper nouns, or spatial modifiers” [1]. Third, one of the mentions cannot contain a number that is not present in the other mention. This sieve would have been an interesting alternative to our NER sieve. A key difference is that the Proper head word match sieve requires the headwords to be equal, while this is not a requirement in our NER sieve. However, similar to the Relaxed string match sieve, the Proper head word match yielded only a small improvement of the CoNLL F1

score in the Stanford sieve model, 0.20 points [1]. Though missing to try this sieve out was a weakness of our project, we do not think that it would have yielded a major improvement of our system performance.

5.6 Ethics and sustainability

In this section, we will reflect on some consequences of our project related to ethics and sustainability.

5.6.1 Ethics

A way in which our system could possibly harm people is through incorrect matches. For instance, consider the following scenario: We have two different persons, John Doe and Jane Doe. They are incorrectly matched by our program because they have identical last names. In that case, there are reasons as to why John Doe could be offended by being mistaken for Jane Doe. It could be offensive to be associated with another person's reputation or doings, for example if John Doe is a private individual and Jane Doe is a politician or a criminal.

A concrete example within the field of digital curation can be used to illustrate this aspect. Coreference resolution can be used as an aid in digital curation by identifying e.g. named entities and replacing all corresponding mentions with the names in question. This could make it easier and more efficient for digital curation workers to review and extract information from documents [4]. However, this could also lead to offensive content if mentions are replaced with the wrong names. Consider the following sentence:

> Doe was sentenced to prison for bank robbery.

Here, ”Doe” refers to Jane Doe. However, if our algorithm resolves ”Doe” incorrectly and considers it a reference to John Doe, the mention replacement will result in the following sentence:

> John Doe was sentenced to prison for bank robbery.

This sentence could be perceived by John Doe as offensive. Therefore, it is important that the results of coreference resolution are evaluated thoroughly and applied cautiously in real-life settings.

5.6.2 Sustainability

Carrying out this computer science project required the use of information and communication technology (ICT). In 2023, it was estimated that between 1.5% and 4% of the greenhouse gas emissions worldwide were caused by the ICT sector [64]. ICT hardware has negative impact on the environment during all phases of its lifecycle: at the beginning of the cycle (manufacturing of the hardware), during the course of the cycle (usage of the hardware), and at the end of the cycle (disposal of the hardware). Manufacturing the hardware requires mining of rare resources. The mining process, in turn, affects the environment, ”leading to pollution of soil, water and air” [65]. Usage of the hardware requires electricity consumption, which also has negative impact on the environment. When the hardware finally is disposed of, it may be ”disposed as e-waste in landfills or disassembled requiring energy” [65], which can lead to environment pollution. The effects of the ICT hardware lifecycle on the environment are referred to as direct effects, which are all negative, while there are also indirect effects connected to the application of ICT, which can be positive as well as negative [64].

During the course of this project, the factor that could be affected the most was considered to be the electricity consumption. Therefore, it was important to minimize the running time of our model. The running time was mainly affected by the efficiency of our program architecture, the efficiency of the external parser, and the amount of data used. The running time for the finished resolution algorithm (including mention detection) using the test data was, in most cases, a few minutes. Parsing the test data took about 30 minutes, but was only executed once in the pipeline.

As we used a rule-based model, our program did not involve a training phase corresponding to that of machine learning or deep learning models. Referring to deep learning models in the field of natural language processing, Strubell, Ganesh, and McCallum [66] state that ”training a state-of-the-art model now requires substantial computational resources which demand considerable energy”. Furthermore, such models may involve the use of several energy demanding specialized components [66].

Based on the reasoning about the ICT hardware lifecycle above, we conclude that the hardware used for this project has indeed had negative impact on the environment, although this impact is difficult to measure in exact numbers. However, with a low running time and using only a laptop for development and execution, we assume that the impact most likely was much less than if a machine learning or deep learning model had been used.

Chapter 6

Conclusions and future work

6.1 Conclusions

For this project, we developed an independent coreference resolution system for German inspired by CoRefGer-rule. The purpose of this project was to answer the following question:

> How will adding a morphological analyzer and semantic information to a rule-based coreference resolution system for German, based on CoRefGer-rule, influence the performance?

Based on the discussion of our results above, we will answer the question by concluding the following:

- **The use of grammatical information obtained from the morphological analyzer was a fundamental and important part of our system.** The morphological analyzer enabled the use of grammatical information about words, such as lemma, part of speech, and gender. Grammatical information was utilized throughout all of our sieves. Thus, it was a fundamental part of our system. It was also an important part, as the use of grammatical information enabled precise comparisons between words and precise formulation of rules. Furthermore, it enabled the implementation of the Pronouns sieve, which otherwise would have been impossible or very difficult to implement. The Pronouns sieve in itself had a big impact on the performance, achieving noteworthy increases of F1 scores by resolving pronouns that otherwise would have been ignored.

- **The semantic information had no significant impact on the performance.** By implementing two different semantic sieves, the GermaNet sieve and the Word vectors sieve, we tried to cover different semantic aspects of words. The purpose of the GermaNet sieve was to match words based on synonymy and hyperonymy/hyponymy relations. In turn, the purpose of the Word vectors sieve was to consider similarity between larger spans of words. However, both sieves failed in considering the semantics of more than a single or a few words, which appeared to be a major weakness. They also yielded very small changes to the F1 scores, increasing as well as decreasing the scores with very few points. All in all, both sieves had a negligible impact on the system performance. We also believe that word vectors are not suitable for use in a rule-based system because of their imprecise nature, as shown in this project.

- **A direct comparison between the performance of our system and CoRefGer-rule is difficult to draw.** We cannot conclude whether or not our system outperformed CoRefGer-rule based on the metric scores, as we identified no clear pattern for these scores. A direct comparison is also complicated by the fact that we built our own system without the use of CoRefGer-rule's source code. However, we can confirm that the morphological analyzer provided our system with relevant and useful information that enabled advanced resolution of pronouns, as concluded above. We see this as a powerful advantage of our system compared to CoRefGer-rule. On the other hand, the semantic sieves had a negligible impact on the performance of our system, which means they provided no obvious advantage of our system.

Lastly, we would like to argue the relevancy of rule-based models like the one developed for this project. The lack of a training phase, equal to that of machine learning and deep learning models, makes rule-based models suitable for low-resource or out-of-domain texts. Furthermore, rule-based models do not necessarily require substantial computational resources, unlike many of the more modern and advanced models, in particular deep learning models.

6.2 Future work

A suggestion for further development of a rule-based coreference resolution model for German news texts is to implement speaker detection. The lack of a

complex sentence analysis was a weakness of our system, because we could not implement speaker detection. By implementing speaker detection, one could implement the Speaker identification sieve, which proved to be a relevant part of the latest version of the Stanford sieve algorithm [1]. Speaker detection was already performed by Krug et al. [7], but their coreference resolution system was tailored to the German literature domain. Therefore, a suggestion is to implement speaker identification in a rule-based coreference resolution system that will be evaluated on German news texts.

Furthermore, one could continue exploring the use of GermaNet. In this project, it proved difficult to utilize the semantic aspect of words. However, we still think that it could be interesting to continue experimenting with GermaNet. The main problem identified with our GermaNet sieve was that it was limited to considering only headwords. To solve part of this problem, one could experiment with adding filters tailored to different kinds of modifiers. One could also try adding other filters based on the different head matching sieves in the original Stanford sieve algorithm [5] [1], such as comparing words of the mention-antecedent pair's clusters.

Another suggestion for further research is to adapt the model of Lee et al. [17] to German. While the Stanford sieve algorithm achieved great performance for English [1], it was eventually outperformed by its successor. The new model combines the strengths of the rule-based sieve algorithm with modern statistical methods from the field of machine learning [17]. To our knowledge, this model has not yet been adapted to German. Therefore, it could be interesting to do so.

References

[1] H. Lee, A. Chang, Y. Peirsman, N. Chambers, M. Surdeanu, and D. Jurafsky, "Deterministic coreference resolution based on entity-centric, precision-ranked rules," *Computational Linguistics*, vol. 39, no. 4, pp. 885–916, Dec. 2013. [Online]. Available: https://www.doi.org/10.1162/COLI_a_00152 [Pages 1, 2, 10, 16, 17, 19, 25, 30, 31, 32, 33, 38, 40, 43, 56, 57, 60, 61, and 65.]

[2] R. Sukthanker, S. Poria, E. Cambria, and R. Thirunavukarasu, "Anaphora and coreference resolution: A review," *Information fusion*, vol. 59, pp. 139–162, 2020. [Online]. Available: https://www.doi.org/10.1016/j.inffus.2020.01.010 [Pages 2, 17, 18, 19, and 20.]

[3] F. Schröder, H. O. Hatzel, and C. Biemann, "Neural end-to-end coreference resolution for German in different domains," in *Proceedings of the 17th Conference on Natural Language Processing (KONVENS 2021)*, K. Evang, L. Kallmeyer, R. Osswald, J. Waszczuk, and T. Zesch, Eds. Düsseldorf, Germany: KONVENS 2021 Organizers, 6–9 Sep. 2021, pp. 170–181. [Online]. Available: https://aclanthology.org/2021.konvens-1.15 [Pages 2, 20, 23, 24, 27, and 49.]

[4] A. Srivastava, S. Weber, P. Bourgonje, and G. Rehm, "Different German and English coreference resolution models for multi-domain content curation scenarios," in *Language Technologies for the Challenges of the Digital Age*, 01 2018. ISBN 978-3-319-73705-8 pp. 48–61. [Online]. Available: https://doi.org/10.1007/978-3-319-73706-5_5 [Pages 2, 22, 27, 28, 37, 39, 49, 56, 57, and 61.]

[5] K. Raghunathan, H. Lee, S. Rangarajan, N. Chambers, M. Surdeanu, D. Jurafsky, and C. Manning, "A multi-pass sieve for coreference resolution," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, H. Li and L. Màrquez, Eds.

Cambridge, MA: Association for Computational Linguistics, Oct. 2010, pp. 492–501. [Online]. Available: https://aclanthology.org/D10-1048 [Pages 2, 19, 23, 34, 36, 40, 43, 56, and 65.]

[6] H. Lee, Y. Peirsman, A. Chang, N. Chambers, M. Surdeanu, and D. Jurafsky, "Stanford's multi-pass sieve coreference resolution system at the CoNLL-2011 shared task," in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, S. Pradhan, Ed. Portland, Oregon, USA: Association for Computational Linguistics, Jun. 2011, pp. 28–34. [Online]. Available: https://aclanthology.org/W11-1902 [Pages 2 and 19.]

[7] M. Krug, F. Puppe, F. Jannidis, L. Macharowsky, I. Reger, and L. Weimar, "Rule-based coreference resolution in German historic novels," in *Proceedings of the Fourth Workshop on Computational Linguistics for Literature*, A. Feldman, A. Kazantseva, S. Szpakowicz, and C. Koolen, Eds. Denver, Colorado, USA: Association for Computational Linguistics, Jun. 2015, pp. 98–104. [Online]. Available: https://www.doi.org/10.3115/v1/W15-0711 [Pages 2, 22, 23, 28, 39, 53, 54, 57, and 65.]

[8] S.-G. Andersson, M. Brandt, I. Rosengren, and I. Persson, *Tysk syntax för universitetsnivå*. Lund: Studentlitteratur, 2002. [Pages 5, 7, 10, 12, 13, 32, and 35.]

[9] J. Meibauer, U. Demske, J. Geilfuß-Wolfgang, J. Pafel, K. H. Ramers, M. Rothweiler, and M. Steinbach, *Einführung in die germanistische Linguistik*. Stuttgart, Weimar: J. B. Metzler, 2007. [Pages xi, 5, 6, 7, and 8.]

[10] U. Klingemann, G. Magnusson, and S. Didon, *Bonniers tyska grammatik*. Lettland: Bonnier utbildning, 2011. [Pages 5, 8, 9, and 10.]

[11] I. Plag, S. Arndt-Lappe, M. Braun, and M. Schramm, *Introduction to English Linguistics*. Berlin, München, Boston: De Gruyter Mouton, 2015. [Online]. Available: https://doi.org/10.1515/9783110378382 [Pages 5, 6, 7, 10, and 13.]

[12] Merriam-Webster Dictionary: Giraffe. Accessed on February 1, 2024. [Online]. Available: https://www.merriam-webster.com/dictionary/giraffe [Page 6.]

[13] Merriam-Webster Dictionary: Spring. Accessed on February 1, 2024. [Online]. Available: https://www.merriam-webster.com/dictionary/spring [Page 6.]

[14] A. Schiller, S. Teufel, C. Stöckert, and C. Thielen, "Guidelines für das Tagging deutscher Textcorpora mit STTS (Kleines und großes Tagset)," 1999. [Online]. Available: http://www.sfs.uni-tuebingen.de/resources/stts-1999.pdf [Pages 6, 11, 14, and 30.]

[15] Cambridge Dictionary: Noun phrases: dependent words. Accessed on February 28, 2024. [Online]. Available: https://dictionary.cambridge.org/grammar/british-grammar/noun-phrases-dependent-words [Page 10.]

[16] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 2024. [Online]. Available: https://web.stanford.edu/~jurafsky/slp3/ [Pages 10, 11, 12, 14, and 15.]

[17] H. Lee, M. Surdean, and D. Jurafsky, "A scaffolding approach to coreference resolution integrating statistical and rule-based models," *Natural Language Engineering*, vol. 23, pp. 1–30, 03 2017. [Online]. Available: https://www.doi.org/10.1017/S1351324917000109 [Pages 11, 20, 53, 57, and 65.]

[18] Universal Dependencies: Universal dependency relations. Accessed on February 28, 2024. [Online]. Available: https://universaldependencies.org/u/dep/ [Pages 12 and 14.]

[19] Cambridge Dictionary: Lemma. Accessed on February 27, 2024. [Online]. Available: https://dictionary.cambridge.org/dictionary/english/lemma [Page 12.]

[20] Britannica: Demonym. Accessed on February 28, 2024. [Online]. Available: https://www.britannica.com/topic/demonym [Page 13.]

[21] The Stanford Natural Language Processing Group: Stanford parser. Accessed on March 8, 2024. [Online]. Available: https://nlp.stanford.edu/software/lex-parser.shtml [Page 14.]

[22] Universal Dependencies: CoNLL-U format. Accessed on March 8, 2024. [Online]. Available: https://universaldependencies.org/format.html [Page 14.]

[23] R. Liu, R. Mao, A. T. Luu, and E. Cambria, "A brief survey on recent advances in coreference resolution," *The Artificial intelligence review*, vol. 56, no. 12, pp. 14 439–14 481, 2023. [Online]. Available: https://www.doi.org/10.1007/s10462-023-10506-3 [Pages 17, 24, 25, and 26.]

[24] Merriam-Webster Dictionary: Anaphora. Accessed on March 1, 2024. [Online]. Available: https://www.merriam-webster.com/dictionary/anaphora [Page 17.]

[25] J. R. Hobbs, "Resolving pronoun references," *Lingua*, vol. 44, no. 4, pp. 311–338, 1978. [Online]. Available: https://www.doi.org/10.1016/0024-3841(78)90006-2 [Page 18.]

[26] S. Lappin and H. J. Leass, "An algorithm for pronominal anaphora resolution," *Comput. Linguistics*, vol. 20, pp. 535–561, 1994. [Online]. Available: https://api.semanticscholar.org/CorpusID:11500985 [Page 18.]

[27] H. H. Mohammadi, A. Talebpour, A. M. Aznaveh, and S. Yazdani, "Review of coreference resolution in English and Persian," 2022. [Online]. Available: https://arxiv.org/abs/2211.04428 [Pages 18, 19, and 20.]

[28] S. E. Brennan, M. W. Friedman, and C. J. Pollard, "A centering approach to pronouns," *Proceedings of the 25th annual meeting on Association for Computational Linguistics*, pp. 155–162, 1987. [Online]. Available: https://www.doi.org/10.3115/981175.981197 [Page 18.]

[29] B. Baldwin, "CogNIAC: high precision coreference with limited knowledge and linguistic resources," in *Proceedings of a Workshop on Operational Factors in Practical, Robust Anaphora Resolution for Unrestricted Texts*, ser. ANARESOLUTION '97. USA: Association for Computational Linguistics, 1997, p. 38–45. [Online]. Available: https://aclanthology.org/W97-1306 [Page 18.]

[30] A. Haghighi and D. Klein, "Simple coreference resolution with rich syntactic and semantic features," in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3 - Volume 3*, ser. EMNLP '09. USA: Association for Computational Linguistics, 2009. ISBN 9781932432633 p. 1152–1161. [Online]. Available: https://aclanthology.org/D09-1120 [Page 19.]

[31] S. Pradhan, L. Ramshaw, M. Marcus, M. Palmer, R. Weischedel, and N. Xue, "CoNLL-2011 shared task: Modeling unrestricted coreference in OntoNotes," in *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task*, S. Pradhan, Ed. Portland, Oregon, USA: Association for Computational Linguistics, Jun. 2011, pp. 1–27. [Online]. Available: https://aclanthology.org/W11-1901 [Page 19.]

[32] H. Telljohann, E. W. Hinrichs, S. Kübler, H. Zinsmeister, and K. Beck, "Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z)," 07 2017. [Pages 20 and 27.]

[33] Universität Tübingen: TüBa-D/Z release 11.0. Accessed on March 18, 2024. [Online]. Available: https://uni-tuebingen.de/en/faculties/faculty-of-humanities/departments/modern-languages/department-of-linguistics/chairs/general-and-computational-linguistics/ressources/corpora/tueba-dz/ [Page 21.]

[34] M. Recasens, L. Màrquez, E. Sapena, M. A. Martí, M. Taulé, V. Hoste, M. Poesio, and Y. Versley, "SemEval-2010 task 1: Coreference resolution in multiple languages," in *Proceedings of the 5th International Workshop on Semantic Evaluation*, K. Erk and C. Strapparava, Eds. Uppsala, Sweden: Association for Computational Linguistics, Jul. 2010, pp. 1–8. [Online]. Available: https://aclanthology.org/S10-1001 [Pages 21 and 22.]

[35] K. Eckart, A. Riester, and K. Schweitzer, "A discourse information radio news database for linguistic analysis," in *Linked Data in Linguistics: Representing and Connecting Language Data and Language Metadata*, C. Chiarcos, S. Nordhoff, and S. Hellmann, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012. ISBN 978-3-642-28249-2 pp. 65–76. [Online]. Available: https://doi.org/10.1007/978-3-642-28249-2_7 [Page 21.]

[36] A. Björkelund, K. Eckart, A. Riester, N. Schauffler, and K. Schweitzer, "The extended DIRNDL corpus as a resource for coreference and bridging resolution," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, Eds. Reykjavik, Iceland: European Language Resources

Association (ELRA), May 2014, pp. 3222–3228. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2014/pdf/891_Paper.pdf [Page 21.]

[37] M. Krug, L. Weimer, I. Reger, L. Macharowsky, S. Feldhaus, F. Puppe, and F. Jannidis, "Description of a corpus of character references in German novels-DROC [Deutsches ROman Corpus]," *DARIAH-DE Working Papers*, vol. 27, pp. 1–16, 2018. [Page 21.]

[38] D. Tuggener, "Incremental coreference resolution for German," PhD thesis, Universität Zürich, Zürich, 2016. [Online]. Available: https://www.doi.org/10.5167/UZH-124915 [Pages 21, 22, 49, and 51.]

[39] M. Klenner, A. Fahrni, and R. Sennrich, "Real anaphora resolution is hard," in *Text, Speech and Dialogue*, P. Sojka, A. Horák, I. Kopeček, and K. Pala, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. ISBN 978-3-642-15760-8 pp. 109–116. [Online]. Available: https://www.doi.org/10.1007/978-3-642-15760-8_15 [Pages 21, 38, 39, 49, 50, 51, 54, and 58.]

[40] SemEval: International workshop on semantic evaluation. Accessed on March 18, 2024. [Online]. Available: https://semeval.github.io/ [Page 22.]

[41] M. Klenner and D. Tuggener, "An incremental entity-mention model for coreference resolution with restrictive antecedent accessibility," in *Proceedings of the International Conference Recent Advances in Natural Language Processing 2011*, R. Mitkov and G. Angelova, Eds. Hissar, Bulgaria: Association for Computational Linguistics, Sep. 2011, pp. 178–185. [Online]. Available: https://aclanthology.org/R11-1025 [Pages 22 and 38.]

[42] D. Tuggener and M. Klenner, "A hybrid entity-mention pronoun resolution model for German using Markov logic networks," in *KONVENS*, Hildesheim, October 2014, pp. 21–29. [Online]. Available: https://www.doi.org/10.5167/uzh-99594 [Page 22.]

[43] A. Björkelund and J. Kuhn, "Learning structured perceptrons for coreference resolution with latent antecedents and non-local features," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, K. Toutanova and H. Wu, Eds. Baltimore, Maryland: Association for Computational

Linguistics, Jun. 2014, pp. 47–57. [Online]. Available: https://www.doi.org/10.3115/v1/P14-1005 [Page 22.]

[44] I. Roesiger and J. Kuhn, "IMS HotCoref DE: A data-driven co-reference resolver for German," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, Eds. Portorož, Slovenia: European Language Resources Association (ELRA), May 2016, pp. 155–160. [Online]. Available: https://aclanthology.org/L16-1024 [Pages 22, 49, and 51.]

[45] M. Dobreva and W. Duff, "The ever changing face of digital curation: introduction to the special issue on digital curation," *Archival Science*, vol. 15, p. 97–100, 2015. [Online]. Available: https://doi.org/10.1007/s10502-015-9243-7 [Page 23.]

[46] M. Krug, "Techniques for the automatic extraction of character networks in German," PhD thesis, Julius-Maximilians-Universität Würzburg, Würzburg, 2020. [Pages 23 and 24.]

[47] T. Gupta, H. O. Hatzel, and C. Biemann, "Coreference in long documents using hierarchical entity merging," in *Proceedings of the 8th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2024)*, Y. Bizzoni, S. Degaetano-Ortlieb, A. Kazantseva, and S. Szpakowicz, Eds. St. Julians, Malta: Association for Computational Linguistics, Mar. 2024, pp. 11–17. [Online]. Available: https://aclanthology.org/2024.latechclfl-1.2 [Page 24.]

[48] M. Vilain, J. Burger, J. Aberdeen, D. Connolly, and L. Hirschman, "A model-theoretic coreference scoring scheme," in *Sixth Message Understanding Conference (MUC-6): Proceedings of a Conference Held in Columbia, Maryland, November 6-8, 1995*, 1995. [Online]. Available: https://aclanthology.org/M95-1005 [Page 25.]

[49] A. Bagga and B. Baldwin, "Algorithms for scoring coreference chains," 1998. [Online]. Available: https://api.semanticscholar.org/CorpusID:14142164 [Page 25.]

[50] X. Luo, "On coreference resolution performance metrics," in *Proceedings of Human Language Technology Conference and*

Conference on Empirical Methods in Natural Language Processing, R. Mooney, C. Brew, L.-F. Chien, and K. Kirchhoff, Eds. Vancouver, British Columbia, Canada: Association for Computational Linguistics, Oct. 2005, pp. 25–32. [Online]. Available: https://aclanthology.org/H05-1004 [Page 26.]

[51] H. Kuhn, "The Hungarian method for the assignment problem," *Naval Res Logist Q*, vol. 2, no. 1–2, pp. 83–97, 1955. [Online]. Available: https://www.doi.org/10.1002/nav.3800020109 [Page 26.]

[52] S. Pradhan, A. Moschitti, N. Xue, O. Uryupina, and Y. Zhang, "CoNLL-2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes," in *Joint Conference on EMNLP and CoNLL - Shared Task*, S. Pradhan, A. Moschitti, and N. Xue, Eds. Jeju Island, Korea: Association for Computational Linguistics, Jul. 2012, pp. 1–40. [Online]. Available: https://aclanthology.org/W12-4501 [Page 26.]

[53] R. Sennrich, G. Schneider, M. Volk, and M. Warin, "A new hybrid dependency parser for German," 01 2009. [Online]. Available: https://www.doi.org/10.5167/uzh-25506 [Page 28.]

[54] R. Sennrich, M. Volk, and G. Schneider, "Exploiting synergies between open resources for German dependency parsing, POS-tagging, and morphological analysis," in *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, R. Mitkov, G. Angelova, and K. Bontcheva, Eds. Hissar, Bulgaria: INCOMA Ltd. Shoumen, BULGARIA, Sep. 2013, pp. 601–609. [Online]. Available: https://aclanthology.org/R13-1079 [Page 28.]

[55] D. Altinok, "Demorphy, german language morphological analyzer," *arXiv preprint*, 2018. [Online]. Available: https://doi.org/10.48550/arXiv.1803.00902 [Page 28.]

[56] H. Schmid, A. Fitschen, and U. Heid, "Smor: A german computational morphology covering derivation, composition, and inflection," in *Proceedings of the IVth International Conference on Language Resources and Evaluation (LREC 2004)*, Lisbon, Portugal, 2004, pp. 1263–1266. [Online]. Available: https://aclanthology.org/L04-1275/ [Page 28.]

[57] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Association for Computational Linguistics (ACL) System Demonstrations*, 2014, pp. 55–60. [Online]. Available: http://www.aclweb.org/anthology/P/P14/P14-5010 [Page 28.]

[58] Universität Hamburg: GermaNER: Free open German named entity recognition tool. Accessed on March 21, 2024. [Online]. Available: https://www.inf.uni-hamburg.de/en/inst/ab/lt/resources/software/germaner.html [Page 28.]

[59] B. Hamp and H. Feldweg, "GermaNet – a lexical-semantic net for German," in *Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, 1997. [Online]. Available: https://aclanthology.org/W97-0802 [Page 28.]

[60] V. Henrich and E. Hinrichs, "GernEdiT – the GermaNet editing tool," in *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, Eds. Valletta, Malta: European Language Resources Association (ELRA), May 2010. [Online]. Available: https://aclanthology.org/L10-1180/ [Page 28.]

[61] C. Fellbaum, "WordNet and wordnets," in *Encyclopedia of Language and Linguistics*, A. Barber, Ed. Elsevier, 2005, pp. 2–665. [Page 29.]

[62] University of Tübingen: GermaNet – an introduction. Accessed on March 25, 2024. [Online]. Available: https://uni-tuebingen.de/en/faculties/faculty-of-humanities/departments/modern-languages/department-of-linguistics/chairs/general-and-computational-linguistics/ressources/lexica/germanet/ [Page 29.]

[63] L. Vällfors, "Coreference resolution for Swedish," Master's thesis, KTH Royal Institute of Technology, 2022. [Pages 30, 31, and 33.]

[64] J. C. Bieser, R. Hintemann, L. M. Hilty, and S. Beucker, "A review of assessments of the greenhouse gas footprint and abatement potential of information and communication technology," *Environmental Impact Assessment Review*, vol. 99, p. 107033, 2023. [Online]. Available: https://doi.org/10.1016/j.eiar.2022.107033 [Page 62.]

[65] B. Krumay and R. Brandtweiner, "Measuring the environmental impact of ICT hardware," *International Journal of Sustainable Development and Planning*, vol. 11, no. 6, pp. 1064–1076, 2016. doi: 10.2495/SDP-V11-N6-1064-1076. [Online]. Available: https://www.doi.org/10.2495/SDP-V11-N6-1064-1076 [Page 62.]

[66] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in NLP," *arXiv preprint*, 2019. [Online]. Available: https://doi.org/10.48550/arXiv.1906.02243 [Page 62.]

TRITA-EECS-EX-2024:206