Master's Thesis in
Computational Linguistics

Generating Questions for German Text

Tobias Kolditz
tklditz@gmail.com

Supervisor: Prof. Dr. Detmar Meurers
Second Examiner: Prof. Dr. Fritz Hamm

Seminar für Sprachwissenschaft
University of Tübingen
November 9, 2015

EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Philosophische Fakultät

Name:

Vorname:

Matrikel-Nummer:

Adresse:

Hiermit versichere ich, die Arbeit mit dem Titel:

_____

im Rahmen der Lehrveranstaltung _____

im Sommer-/Wintersemester _____ bei _____

selbständig und nur mit den in der Arbeit angegebenen Hilfsmitteln verfasst zu haben.

Mir ist bekannt, dass ich alle schriftlichen Arbeiten, die ich im Verlauf meines Studiums als Studien- oder Prüfungsleistung einreiche, selbständig verfassen muss. Zitate sowie der Gebrauch von fremden Quellen und Hilfsmitteln müssen nach den Regeln wissenschaftlicher Dokumentation von mir eindeutig gekennzeichnet werden. Ich darf fremde Texte oder Textpassagen (auch aus dem Internet) nicht als meine eigenen ausgeben.

Ein Verstoß gegen diese Grundregeln wissenschaftlichen Arbeitens gilt als Täuschungs- bzw. Betrugsversuch und zieht entsprechende Konsequenzen nach sich. In jedem Fall wird die Leistung mit **„nicht ausreichend“ (5,0)** bewertet. In schwerwiegenden Fällen kann der Prüfungsausschuss den Kandidaten/die Kandidatin von der Erbringung weiterer Prüfungsleistungen ausschließen; vgl. hierzu die Prüfungsordnungen für die Bachelor-, Master-, Lehramts- bzw. Magisterstudiengänge.

Datum: _____ Unterschrift: _____

Abstract

This thesis contributes to the limited amount of research that went into automatic question generation for German by developing and implementing a transformation-based question generation system for German text. The system dynamically generates syntactic questions asking for information that is explicitly encoded in a text. It operates on syntactic representations of declarative sentences enriched with morphological features, lexical semantic categories and antecedents of pronouns. The system architecture builds on previous work for English, especially the factual question generation system by Heilman (2011), but also incorporates a number of new language-specific components dealing with the mapping from answer phrases to question phrases, word order and agreement. Questions generated from three newswire texts were evaluated manually for grammaticality, acceptability, specificity and informativeness. Most questions were at least acceptable, roughly half of them specific and informative with regard to the requested information. Parser errors and context-dependent words have been identified as the main reasons behind low-quality questions.

Zusammenfassung

Diese Arbeit leistet einen Beitrag zur bisher recht überschaubaren Forschung, die sich mit der automatischen Generierung von Fragen für das Deutsche beschäftigt. Dazu wird ein transformationsbasiertes System zur Generierung von Fragen für deutschsprachige Texte entwickelt und implementiert. Das System erzeugt syntaktische Fragen, die explizit im Text kodierte Informationen erfragen. Es operiert dabei auf syntaktischen Repräsentationen von Deklarativsätzen, die mit morphologischen, lexikalisch-semantischen und Koreferenz-Informationen angereichert sind. Die Systemarchitektur baut auf Vorarbeiten für das Englische auf, vor allem auf das System von Heilman (2011) zur automatischen Generierung von *factual questions*, enthält dazu aber auch eine Reihe neuer Komponenten, die speziell für die Abbildung von Antwortphrasen auf Fragephrasen sowie die Modellierung von Wortstellung und Kongruenzphänomenen im Deutschen entwickelt wurden. Auf Grundlage dreier Agenturmeldungen erzeugte Fragen wurden manuell auf Grammatikalität, Akzeptabilität, Spezifität und ihren Informationsgehalt hin untersucht. Die meisten Fragen waren zumindest akzeptabel, etwa die Hälfte aller Fragen waren dazu spezifisch und informativ im Hinblick auf die erwartete Antwort. Parserprobleme und kontextabhängige Wörter konnten als Hauptursachen für Fragen von geringer Qualität ausgemacht werden.

Contents

List of Tables **vi**

List of Figures **viii**

1. Introduction **1**

2. Background **3**

2.1. Previous Systems . . . 3
2.1.1. Motivation and Strategy . . . 3
2.1.2. Input . . . 5
2.1.3. Output . . . 5
2.1.4. Method . . . 6
2.1.5. Question Generation for German . . . 12
2.2. Evaluation Schemes . . . 13
2.3. Question Taxonomy . . . 15
2.4. General Challenges . . . 17
2.5. Challenges Specific to German . . . 19

3. Automatic Linguistic Annotation **22**

3.1. Basic Linguistic Units . . . 22
3.2. Part-of-speech Tags . . . 22
3.3. Constituency Parse . . . 24
3.4. Morphologically Rich Tags and Lemmas . . . 25
3.5. Dependency Information . . . 26
3.5.1. Lexical Heads . . . 26
3.5.2. Grammatical Functions . . . 27
3.6. Semantic Classes and Groups . . . 27
3.7. Coreference . . . 29
3.8. Data Structures . . . 30

4. Question Generation **31**

4.1. Identifying Potential Answer Phrases . . . 31
4.1.1. Redundant Rules . . . 32

4.1.2. Restrictions of Non-Syntactic Nature . . . 33
4.1.3. Extractions from Finite Subordinate Clauses . . . 33
4.2. Generating Question Phrases from Answer Phrases . . . 36
4.2.1. Nominal Phrases . . . 37
4.2.2. Prepositional Phrases . . . 42
4.2.3. Subordinate Clauses . . . 43
4.2.4. Other Units . . . 55
4.3. Undoing Topicalization . . . 57
4.3.1. Unmarked Word Order . . . 58
4.3.2. Features of Constituent Types . . . 62
4.3.3. Implementation . . . 64
4.3.4. Transitive Rules . . . 65
4.4. Resolving Coreferences . . . 73
4.5. Inflection . . . 75
4.5.1. Reverse Lemmatizer . . . 75
4.5.2. Inflecting Verbs, Antecedents and Possessive Pronouns . . . 76
4.6. Post-Processing . . . 76

5. Evaluation **77**
5.1. Data . . . 77
5.2. Results . . . 78
5.2.1. Quality Rating and Characteristic Examples . . . 79
5.2.2. Quantitative Results . . . 82
5.2.3. Error Analysis . . . 83
5.3. Coreference Resolution Experiment . . . 88
5.4. Discussion . . . 90

6. Conclusions and Future Work **92**

A. Tags and Labels **102**
A.1. Small STTS (1999) . . . 102
A.1.1. Original Set . . . 102
A.1.2. TIGER Modifications . . . 104
A.2. Large STTS (1999) . . . 105
A.3. Node Labels . . . 105
A.4. Edge Labels . . . 106

B. Rules **108**
B.1. Heilman's (2011) `Tregex` Movement Constraints . . . 108
B.2. Answer-Question Phrase Mapping for PPs . . . 108

List of Tables

2.1. Example for a pattern matching an input sentence and selecting parts of it to fill the variables in a question template (Wyse & Piwek, 2009, p. 72) 10
2.2. Example for learning and validating a strong lexico-syntactic pattern (Curto, Mendes, & Coheur, 2012, pp. 154–158) 12
2.3. Question quality according to the content of the requested information, first proposed by Graesser, Langston, and Lang (1992), adapted from Graesser and Person (1994) 16

3.1. Entity types recognized by Faruqui and Padó (2010) 28
3.2. Sets of semantic classes that make up semantic groups 29

4.1. Tregex patterns marking constituents as unmovable 32
4.2. Question phrases for nominative noun phrases 38
4.3. Question phrases for accusative noun phrases 39
4.4. Heuristics for determining temporal NP subgroups 41
4.5. Question phrases for dative and genitive noun phrases 41
4.6. Question phrases for prepositional phrases with *auf* as head 42
4.7. Types of adverbial clauses 48
4.8. Mapping from subordinating conjunctions to question phrases 54
4.9. Mapping from feature triples to relative order scores 65

5.1. Number of questions per answer type (category and function) 78
5.2. Question quality scores, ordered from worst to best 79
5.3. Relative frequencies of quality scores (1–6) 82
5.4. Absolute frequencies of quality scores and question phrases 83
5.5. Codes for different types of errors 84

A.1. Small STTS (Schiller, Teufel, Stöckert, & Thielen, 1999) with additional tags SGML and SPELL (Albert et al., 2003, Appendix B) 104
A.2. Linguistic attributes, their values and the STTS-tags they apply to (Schiller et al., 1999, p. 8) 105

A.3. Constituent labels in the NEGRA and TIGER treebank, compiled from `http://www.coli.uni-saarland.de/projects/sfb378/negra-corpus/knoten.html` and Albert et al. (2003) 106
A.4. Edge labels according to the annotation schemes of NEGRA and TIGER, compiled from `http://www.coli.uni-saarland.de/projects/sfb378/negra-corpus/kanten.html` and Albert et al. (2003) 107

List of Figures

2.1. Text-to-text question generation trapezoid based on Vauquois' triangle for rule-based machine translation, adapted from Piwek and Boyer (2012, p. 5) 7

3.1. Stanford parse based on tags from the Stanford tagger 23
3.2. Stanford parse based on tags from the TreeTagger 24
3.3. Dependency parse with number of transitive dependents for each word 27

4.1. Stanford parses for the first four sentences in (44) 53

5.1. Absolute frequencies of different types of errors 85

1. Introduction

Automatically generating questions is an interesting task both from a practical and theoretical point of view. A variety of question generation systems have been developed, mostly for educational purposes, such as the acquisition of a foreign language or academic writing skills, the assessment of vocabulary knowledge or reading comprehension and automatic tutoring with interactive dialogues (cf. Le, Kojiri, & Pinkwart, 2014). Some systems enhance the performance of interactive question answering (Harabagiu, Hickl, Lehmann, & Moldovan, 2005; Yao, Tosch, et al., 2012) or explore the computational and linguistic possibilities for question generation (QG) without a specific application in mind. These possibilities include statistical methods for selecting interesting question content and ranking output according to quality, machine learning models for the classification of question types and hand-crafted or automatically extracted rules and patterns operating on different levels of linguistic abstraction to create questions based on declarative sentences, texts or structured data.

While a lot of systems deploying various methods for different purposes have been developed for the English language, there is very little research on question generation for German – in fact, I was only able to find a template-based system that focuses on the selection of important concepts in a text rather than question construction (see section 2.1.5). The computational linguistic problems related to dynamically generating syntactic questions in German have not been addressed so far. This thesis takes a first step in filling this gap by implementing a transformation-based QG system[1] for German[2]. The system takes text as input, performs a linguistic analysis using two parsers, a morphological tagger, a lexical-semantic net and a coreference resolution system, selects potential answer phrases (NPs, PPs and embedded clauses), replaces them with matching question phrases and transforms the syntactic representations of the respective declarative sentences into questions. All information necessary for answering a question should be contained in the input text. The focus of this thesis is on the computational linguistic challenges of question generation for German, that is, on the

[1]The system is implemented in roughly 5000 lines of Java code and additional files encoding linguistic knowledge. It can be accessed via this link: `https://drive.google.com/folderview?id=0ByFGXcLQeCGXN1RxVS1RMlFHS0U&usp=sharing`. For legal reasons, not all external dependencies are included. If you are interested in running the code and have problems installing the dependencies, please contact the author.

[2]Or more precisely, the German written standard variety of German.

combination of challenging language-specific phenomena (a complex system of question words, the interaction of word order and information structure, overt morpho-syntactic and semantic agreement) and restricted computational linguistic means.

After a brief systematic overview of prior work and an outlook on the challenges of question generation, I will present the details of the automatic linguistic annotation and the core of the system concerned with the generation of questions. An evaluation on a small set of texts will reveal different types of errors and their frequencies and will give us a first indication of the overall performance of the system.

2. Background

Before we dive into the details of the system developed in this thesis, let us have a look at the existing literature on the topic. First, I try to give an overview of different approaches to question generation and point out some problems with evaluations. Section 2.3 briefly discusses a popular question taxonomy. The last two sections of this chapter are devoted to the general and German-specific challenges of question generation.

2.1. Previous Systems

Although there has been less research on question generation than on the related problem of question answering, quite a number of different QG systems have been developed, especially in the last ten years. Rus, Cai, and Graesser (2008) give the following definition of question generation:

> "As a first approximation, we define Question Generation as the automatic generation of questions (Factual questions, Yes/No-questions, Why-questions, etc.) from inputs such as text (in particular, declarative sentences), raw data, and knowledge bases." (Introduction, para. 1)

The definition already points out two dimensions along which QG systems may vary, namely the *input* (texts, raw data or knowledge bases) and the *output* (different types of questions). Another important point is, of course, "the relation between the input and the output" (Piwek & Boyer, 2012, p. 3) and more specifically, the way the output is generated based on the input, which I will simply refer to as the *method*. But before I come to the input, the output and the method, I will discuss different *motivations* and *strategies* behind QG systems.

2.1.1. Motivation and Strategy

With respect to the motivations behind different systems, their purposes or goals, we can broadly distinguish between QG systems that were developed for a specific application and application-neutral systems. Application-neutral systems can be task-oriented or explorative. Examples for the former are the systems that participated in the first *Question Generation Shared Task Evaluation Challenge (QGSTEC'10)*, which challenged

participants to generate either "a list of 6 questions from a given input paragraph" that "should be at three scope levels: 1 x broad (entire input paragraph), 2 x medium (multiple sentences), and 3 x specific (sentence or less)" (Task A; Rus et al., 2010, pp. 47f.) or two questions of a given type (*who*, *where*, *when*, *which*, *what*, *why*, *how many*, *how long* or *yes/no*) for a single sentence (Task B; Rus et al., 2010, pp. 51ff.). However, strategy-wise there is no clear line separating task-based and explorative systems – many systems in the QGSTEC'10 actually follow an explorative approach in that they choose a certain method (relying more on syntax (Ali, Chali, & Hasan, 2010) or semantics (Yao & Zhang, 2010)) based on which they try to generate as many different questions as possible. This bottom-up strategy is characteristic for most application-neutral systems: The authors of these systems explore the possibilities of new computational methods or linguistic features (e.g., semantic representations (Yao, Bouma, & Zhang, 2012; Yao & Zhang, 2010), discourse cues (Agarwal, Shah, & Mannem, 2011) or Latent Dirichlet Allocation for identifying subtopics in texts, which can be used to rank generated questions according to topic relevance (Chali & Hasan, 2012, 2015)) or apply and extend existing methods to new languages, such as Basque (Aldabe, de Lacalle, Maritxalar, Martinez, & Uria, 2006; Aldabe, Maritxalar, & Soraluze, 2011), French (Bernhard, De Viron, Moriceau, & Tannier, 2012), Punjabi (P. Garg & Bedi, 2013; S. Garg & Goyal, 2013), Hindi (Kaur & Bathla, 2015) and German (this thesis). Systems developed for a specific application, on the other hand, usually follow a top-down strategy: They need to fulfill a given task and choose all necessary means accordingly.

Most application-specific systems are developed for educational purposes. Le et al. (2014, pp. 326) distinguish three subclasses of educational systems: systems for knowledge/skills acquisition, knowledge assessment and tutorial or 'Socratic' dialogues. Knowledge or skills acquisition QG systems, as the name suggests, support learners in the acquisition of a certain skill or knowledge, e.g., grammar knowledge and conversation skills in a foreign language (Kunichika, Katayama, Hirashima, & Takeuchi, 2001), reading comprehension strategies (Chen, Aist, & Mostow, 2009; Mostow & Chen, 2009), academic writing (Liu, Calvo, & Rus, 2012, 2014), knowledge about history via self-directed learning (Jouault & Seta, 2013, 2014; Jouault, Seta, & Hayashi, 2015) or argumentation skills (Le et al., 2014; Le & Pinkwart, 2015). Systems of the second subclass help assessing the knowledge of learners, e.g., vocabulary knowledge (Brown, Frishkoff, & Eskenazi, 2005; Mitkov, An Ha, & Karamanis, 2006; Susanti, Iida, & Tokunaga, 2015)[1] and factual knowledge after reading a text (Heilman, 2011; Heilman & Smith, 2009, 2010). The last subclass of educational QG systems generates questions to create dialogues. Le et al. (2014) mention three systems that generate question for

[1]Some of these systems generate different kinds of cloze test items (e.g., multiple-choice cloze questions) instead of questions in the narrow, linguistic sense, see section 2.1.3.

tutorial dialogues (Graesser et al., 2008; Lane & VanLehn, 2005; Olney, Graesser, & Person, 2012), but the distinction between dialogues for educational and other purposes is not always very clear, as dialogues can serve multiple purposes at once. For example, THE-MENTOR by Curto et al. (2012) is designed to improve the virtual agent of the FalaComigo project in its interaction with tourists in Portugal. Curto et al. (2012, p. 147) themselves describe the purpose of the system as a mixture of education and entertainment. Other projects working on automatic dialogue generation take a more general approach and do not specify one single purpose, for example, the CODA project (Coherent Dialogue Automatically Generated from Text) whose question generation system is described in Piwek and Stoyanchev (2010). Another application for automatic question generation in interactive contexts is the enhancement of question answering systems by building a database of predicted question-answer pairs (Harabagiu et al., 2005; Yao, Tosch, et al., 2012).

2.1.2. Input

The overwhelming majority of QG systems takes some amount of text as input. This can be a sentence (as in Task B of the QGSTEC'10), a paragraph (as in Task A of the QGSTEC'10) or a whole collection of texts (Harabagiu et al. (2005) generate questions from a collection of topic-related documents to populate a database with question-answer pairs in order to improve their interactive question answering system FERRET). But, as indicated by the definition above, the input to a QG system need not always be text – questions may also be generated based on some kind of semantic representation or a database: Jouault and Seta (2013) use patterns to generate questions based on two concept maps, one built by the user of the system and another one built by the system from two databases (with concepts and relations extracted from Wikipedia) and the user's concept map. Theoretically, questions could also be generated from a simple relational database. Different types of input usually go along with different motivations and different methods.

2.1.3. Output

The examples of automatically generated questions in the definition above (factual questions, yes/no-questions and why-questions) all belong to what Piwek and Boyer (2012) call *syntactic questions*. They conclude that this category is too narrow to capture the output of all systems that are usually described as QG systems and propose the notion of a *pragmatic question*, "i.e., a request for information" (Piwek & Boyer, 2012, p. 3) instead. The focus of this thesis is on syntactic questions and the computational linguistic problems that arise when we try to generate them automatically from a

German text, but it is worth noting that some systems, especially those developed for the purpose of knowledge assessment, generate questions that are only captured by Piwek and Boyer's (2012) pragmatic notion. These systems usually generate different kinds of cloze test items that prompt the user to select one of several possible answers or freely choose an answer, for example, multiple-choice cloze items (in the early version of Project LISTEN's Reading Tutor (Mostow, Tobin, & Cuneo, 2002)), single choice and completion exercises (in the bilingual system by Gütl, Lankmayr, Weinhofer, and Höfler (2011)) and wordbank questions (for vocabulary testing in Brown et al. (2005)). To support the acquisition and assessment of knowledge in certain formalizable domains, some systems also generate non-syntactic questions in formal languages, e.g., algebra problems (Singh, Gulwani, & Rajamani, 2012) or questions about high school mechanics using first-order logic (Singhal, Henz, & Goyal, 2015).

2.1.4. Method

The question generation process of different systems can be broken down into a sequence of abstract steps[2]:

1. content selection
2. question type identification
3. question construction
4. output ranking or selection

In the first step, a QG system selects parts of the input that can be used to generate interesting or useful questions. After that, the question type is identified based on the selected material. In a third step, the question is constructed from the content according to the identified question type. If the system overgenerates, it may utilize some ranking or selection algorithm to downvote or exclude ill-formed questions. Not all steps need to be implemented in each system and different systems focus more or less on certain steps, according to their motivation, input and output[3]. For example, the above-mentioned systems that generate cloze test items for vocabulary and reading comprehension assessment focus almost exclusively on content selection (and the generation of good distractors, with which I am not concerned here), whereas sentence-based question systems that generate syntactic questions usually do not care too much about content selection (they only select linguistically plausible answer phrases) and focus more on question type identification and question construction.

[2]Piwek and Boyer (2012, p. 2) cite the first three steps from the question generation website `questiongeneration.org`, which seems to be no longer available [October 13, 2015].

[3]And according to the background of the people involved in developing the system. Authors with a psychological or educational background tend to focus more on content selection, whereas (computational) linguists are usually more interested in steps two and three.

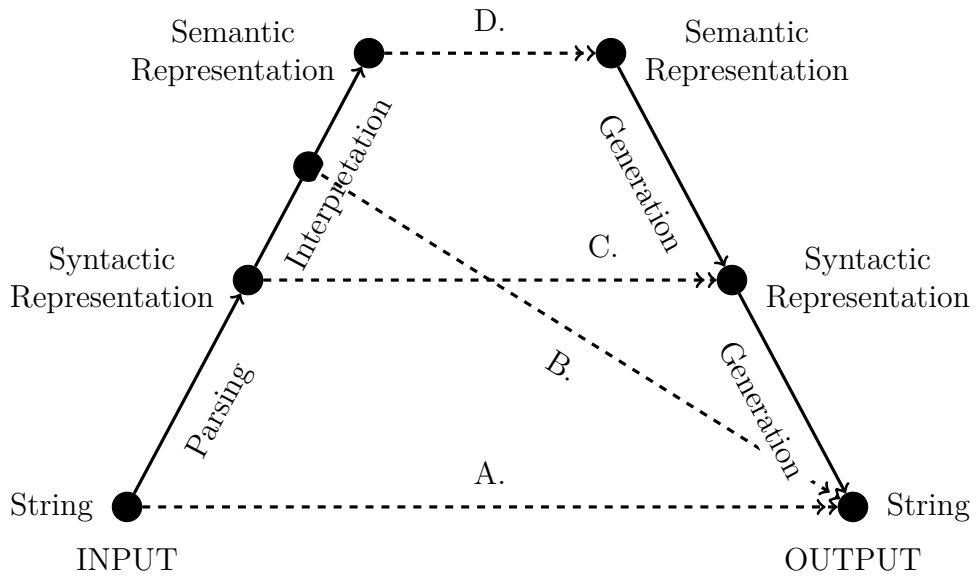

Figure 2.1.: Text-to-text question generation trapezoid based on Vauquois' triangle for rule-based machine translation, adapted from Piwek and Boyer (2012, p. 5).

Piwek and Boyer (2012) try to establish a hierarchy of text-to-text question generation approaches in the form of a trapezoid based on Vauquois' triangle for rule-based machine translation, which I copied in Figure 2.1. They classify different question generation methods based on their representations of input and output and the way they transforms one into the other. At the bottom, we have a simple string-to-string transformation (2.1A), a system following this method would generate a question without any linguistic modeling – none of the systems mentioned in this chapter fall under this category. Further up the trapezoid, they locate systems that operate on syntactic or semantic representations (or a mixture of both). These systems either transform the input representation to an output representation from which then the output string is generated (2.1C, 2.1D) or generate the output string directly from the input representation (2.1B), usually with the help of certain patterns or templates. However, this classification has some weak points: It is only applicable for systems that take single sentences as input, not others that operate on paragraphs, collections of texts, semantic representations or databases; it ignores content selection (probably because content selection is not really important on a sentence level, as mentioned above) and output ranking; and it oversimplifies a bit in that it gives the impression that systems only perform some syntax- or semantics-based transformations, while usually many heterogeneous features are involved in identifying the question type and constructing a question.

The following overview classifies QG systems according to their most characteristic feature[4] and focuses on computational linguistic problems, that is, steps three and four.

[4]The choice is of course disputable and a number of other classifications would be equally valid.

Syntax-Based Transformations

Most of the QG systems for single sentences and many other text-based systems rely (at least partly) on syntactic representations. These representations are either used directly to generate a question string (cf. Figure 2.1B) or transformed into a second intermediate representation from which the output is generated (cf. Figure 2.1C). The first approach works with rules that are informed by syntactic parses but do not operate on them: Potential answer phrases are selected based on the syntactic parse. The appropriate question word for each answer phrase is identified from its part of speech, grammatical function and additional semantic information in the form of lexical semantic types (Kalady, Elikkottil, & Das, 2010; Kunichika et al., 2001; Varga & Ha, 2010)[5] and semantic roles (Pal, Mondal, Pakray, Das, & Bandyopadhyay, 2010). The answer phrase is replaced by the identified question phrase and the declarative sentence is converted to a question by some shallow syntax-based transformations (such as subject-auxiliar inversion and the introduction of do-support, if necessary).

Systems following the second approach select potential answer phrases and identify question words in the same way as the above systems, but to convert a declarative sentence into a question, they perform transformations in the narrow, syntactic sense and generate the string from a second intermediate representation. This method was first deployed by Gates (2008), extended by Heilman and Smith (2009, 2010) and Heilman (2011) and later adopted by Bernhard et al. (2012) for French. The system developed in this thesis follows the second approach – each question is generated from its fully specified syntactic tree (which, in addition, holds morphological, lexical semantic and discourse-related information, see chapter 3).

Semantic Transformations

A system that follows path D in Figure 2.1 was implemented by Yao and Zhang (2010). Similar to the previously mentioned sentence-based systems, they identify potential answers based on lexical semantic information (using a named entity recognizer and an ontology). After that, they semantically parse the sentences with a parser that outputs representations in *Minimal Recursion Semantics*. From these representations, they extract simpler statements. The semantic representations of these simpler statements are then transformed, which in this case simply means that elementary predications for previously identified potential answer phrases are replaced with elementary predications for question words. Questions are generated from these transformed representations with an existing language generation tool. Sometimes, this tool generates multiple questions from one representation. To select the best questions, Yao, Bouma, and Zhang (2012)

[5] Kalady et al. (2010) also deploy a statistical content selection approach with template-based question construction for definitional questions, see below.

add a ranking module that deploys a statistical language model trained on a corpus of questions.

Yao, Bouma, and Zhang (2012, pp. 36f) mention several advantages of their approach compared to the syntax-based approach. The main point is that a semantics-based system allows for a more general, modular approach: Most of the question construction work can be transfered to an independent language generation module, which avoids hand-crafting complicated transformation rules that ensure grammatical output. The QG-specific part of the system is reduced to the selection of potential answer phrases and the generation of appropriate question phrases. The language generation component can be developed and optimized independently of question generation and may allow for any degree of syntactic variation the language allows for a given semantic configuration (Yao, Bouma, and Zhang (2012) mention diathetic variation and different argument realizations in English[6]).

A more shallow semantic approach was first proposed by Mannem, Prasad, and Joshi (2010) and later adopted by Chali and Hasan (2012, 2015): Both generate questions based on semantic role label parses. Their approach, although semantic in nature, does not share the advantages of Yao, Bouma, and Zhang's (2012) system, since they could not use any independent language generation module, but had to design their own transformation rules[7] for question formation.

Manually Created Patterns and Templates

The term *pattern* in the literature is used to refer to a description of a certain linguistic configuration that is matched against the input to select certain sentences or to refer to a question *template*, which usually is a sentence with some open variables that need to be replaced by parts of the input. Both can occur together, for example, Wyse and Piwek (2009) use `Tregex` patterns to select sentences and mark phrases for which a predefined question template exists, see Table 2.1. Templates sometimes also incorporate transformation rules, e.g., the question template for the citation category 'opinion' in Liu et al. (2012, p. 112), copied in (1-a), which for (1-b) leads to the questions in (1-c), both taken from Liu et al. (2012, p. 103).

(1) a. Why +subject_auxiliary_inversion()? What evidence is provided by +subject+ to prove the opinion? Do any other scholars agree or disagree with +subject+?

[6] Although we should keep in mind that all these variations need to be implemented somehow in the generation component. That Yao, Bouma, and Zhang (2012) do not have to deal with these problems is due to the fact that they outsource this work to an existing language generation module, not due to their use of semantic representations per se.

[7] Chali and Hasan (2015) mention a "set of 350 general-purpose rules [...] to transform the semantic-role labeled sentences into the questions" (p. 8).

Pattern	`NP < personNP=g1 . (VBN=g2 . (as . (a . NN=g3)))`
Match	[1]Emmanuel-Joseph Sieyès -LRB- 1748 – 1836 -RRB- [2]trained as a [3]priest and became assistant to a bishop.
Question Template	`What did /1 /2->VPAST as?`
Question	What did Emmanuel-Joseph Sieyès train as?
Answer Template	`a /3`
Answer	a priest

Table 2.1.: Example for a pattern matching an input sentence and selecting parts of it to fill the variables in a question template (Wyse & Piwek, 2009, p. 72).

b. Cannon (1927) challenged this view mentioning that physiological changes were not sufficient to dicriminate emotions.

c. Why did Cannon challenge this view mentioning that physiological changes were not sufficient to discriminate emotions? What evidence is provided by Cannon to prove the opinion? Does any other scholar agree or disagree with Cannon?

This mixed approach is very similar to the shallow transformations mentioned in the section on syntax-based transformations.

There are several reasons why some systems use question templates instead of transformations to construct questions. Without any evaluation and only judging by the combinatorial possibilities for error, it is safe to assume that template-based questions have a higher probability of being linguistically well-formed, for example, if we look at the question template in Table 2.1, the system might mistakenly select a non-person NP or an unexpected verb, whereas a transformation based system additionally may generate an ungrammatical word order or may include too much or vague material from the input. While the template-based approach is also used on the sentence level (e.g., in the above-mentioned system by Wyse and Piwek (2009)), it is particularly useful if questions with wide scope (a paragraph or a whole text) are to be generated: Mostow and Chen (2009) infer a situation model from modal verbs in a text based on which they generate *what*, *why* and *how* questions; Kalady et al. (2010) generate definitional questions from what they call *Up-Keys*, terms marked by a named entity recognizer which have a high term frequency and occur at the beginning, at the end and throughout the whole document (for details, see Das & Elikkottil, 2010), i.e. terms that are assumed to be central or more important than other terms based on distributional properties. Since the answers to these questions may span more than one sentence, a transformational approach is not useful in these cases. A third case for templates is the generation of questions in narrow domains, where only a small set of predefined question types is required: Liu et al. (2012, 2014) define six question templates, one for each

citation category (*opinion* (see (1-a)), *result*, *system*, *application*, *method*, *aim*) – these six templates ensure well-formed questions in most cases but preclude any variation. Lastly, templates are useful if no linguistic material is available from the input of the QG system, which may be the case if the input is a database or a concept map (Jouault & Seta, 2013), or if textual input is transformed into such a structure before question generation (Olney et al., 2012), allowing for questions of varying scope and specificity.

Lexico-Syntactic Patterns

While most transformation rules, patterns and templates for question generation are created manually, Curto et al. (2012) present an interesting approach to automatically generate patterns with the help of web queries.

From a set of question-answer pairs, they first automatically select what they call *seed/validation pairs*, two question-answer pairs that are the same with respect to their questions' syntactic structure, the questions' first constituent (the question phrase) and the category of the answer. Based on the first question-answer pair (the seed pair), the system builds queries "from the permutations of the set composed by: (a) the content of the phrase nodes (except the *Wh*-phrase) of the question, (b) the answer and (c) a wildcard (*)" (Curto et al., 2012, p. 154). From a web search engine, they retrieve sentences that match the query (they first retrieve candidates matching on a string-level and then parse these sentences to select those with matching phrase nodes). Then, they extract a pattern from each matching sentence, consisting of the phrase node values of the material given in the question or the answer and the lexical element that matched the wildcard. For a concrete example, see the first half of Table 2.2, which summarizes and extends an example given by Curto et al. (2012, pp. 154–158): Here, two different queries are generated by permuting the phrase nodes of the seed question-answer pair and the wildcard symbol. For each permutation, the table gives one example of a retrieved sentence (or sentence fragment) which matches on the string level and with respect to the syntactic parse, together with a pattern extracted from each of the two sentences.

In a second phase, the extracted patterns are validated against the validation pair: A query is formed by instantiating the pattern with material from the validation question and answer. The number of documents returned by the query is normalized by the maximum number of documents returned by any query – if this score exceeds a certain threshold, the pattern is validated. The validation queries for the two patterns in the example in Table 2.2 returned one and 216 documents respectively[8]. We do not know about the threshold and what the maximum number of results would have been, but it is safe to assume that the first pattern failed to reach the threshold. The patterns in

[8]Google search, October 14, 2015.

Learning	Seed pair	*Who sculpted the statue of David? – Michelangelo* [WHNP Who] [VBD sculpted] [NP the statue of David] – [NP Michelangelo]
	Queries	`"Michelangelo * sculpted the statue of David"` `"the statue of David sculpted * Michelangelo"`
	Matching sentences	*Michelangelo has sculpted the statue of David* *the statue of David sculpted by Michelangelo*
	Learned Patterns	`NP{ANSWER} [has] VBD NP` `NP VBD [by] {ANSWER}`
Validation	Validation pair	*Who invented penicillin? – Alexander Fleming* [WHNP Who] [VBD invented] [NP penicillin] – [NP Alexander Fleming]
	Instantiations (= queries)	`"Alexander Fleming has invented penicillin"` `"penicillin invented by Alexander Fleming"`
	Number of results (Google)	1 for the instantiation of `NP{ANSWER} [has] VBD NP` 216 for the instantiation of `NP VBD [by] {ANSWER}`

Table 2.2.: Example for learning and validating a strong lexico-syntactic pattern (Curto et al., 2012, pp. 154–158).

the example are *strong* patterns, which "are built by forcing every phrase (except the *Wh*-phrase) to be present in the patterns" (Curto et al., 2012, p. 156). To account for different verb forms and allow for more variation, they also generate *inflected* and *weak* patterns, which are ignored here.

In order to generate questions based on validated patterns, Curto et al. (2012) look for matching structures in a target text. If a match is found, the question is generated following the model of the seed question (the question phrase is added and the remaining phrases are aligned and reordered).

2.1.5. Question Generation for German

As far as I know, the only QG system for German is the *Enhanced Automatic Question Creator (EAQC)* developed by Gütl et al. (2011). The EAQC generates single choice, multiple choice and completion exercises as well as open-ended questions from both English and German text. The first three output types are only questions in the pragmatic sense: Multiple choice and completion exercises are cloze tests with or without a list of possible completions; single choice exercises are statements that are supposed to be rated true or false. Gütl et al. (2011) focus on the extraction of relevant concepts from a text (and the generation of distractors and reference answers), that is, the content selection step, and say very little on how questions are constructed. Thus, their approach is complementary to the one followed in this thesis, where the focus is exclusively on question type identification and question construction. To construct

open-ended questions, they use "several patterns depending on the special annotation type in the selected concept" (Gütl et al., 2011, p. 29). They do not give any details about these patterns, but by looking at their examples, we can infer that these are rather general templates. In (2), I repeat their examples for open-ended questions.

(2) A "good" and a "bad" open-ended question (Gütl et al., 2011, p. 34)

a. What do you know about Modern NLP algorithms in the context of Natural language processing?

b. What do you know about Natural Language processing in the context of Natural language processing?

Apparently, the template is *What do you know about X in the context of Y*, where X is a specific topic from domain Y. The task for the EAQC is to select important concepts, determine their type and the relation between them and, if possible, substitute them into a predefined template.

2.2. Evaluation Schemes

Due to the big variety of question generation systems in terms of motivation, input, output and methods, it is very difficult to compare their results. And even for similar systems, there are a number of problems, which I will discuss in this section. On a high level, we can distinguish between intrinsic and extrinsic evaluations. Extrinsic evaluations are usually preferred for application-specific systems, since they measure the quality of a system in an authentic setting. However, extrinsic evaluations, by their nature, are task specific, cannot be easily compared and are not an option for application-neutral systems. This is why this section only discusses intrinsic evaluations.

The most influential intrinsic evaluation scheme is the one that was used in the QGSTEC'10. Human judges were supposed to rate questions according to five criteria: relevance, question type, syntactic correctness and fluency, ambiguity and variety (Rus et al., 2010, p. 53). For each criterium, questions received a score from 1 to 2, 3 or 4. For all questions generated by a system, average scores are computed for each criterium. Systems that achieve lower average scores are better than systems with higher average scores. In the QGSTEC'10 the system by Varga and Ha (2010) scored "best on all criteria except for 'Variety'" (Rus et al., 2012, p. 198), where it was outperformed by all other systems. When a penalty was given for missing questions the semantics-based system by Yao and Zhang (2010) performed best in all categories. The criteria defined in this scheme were also used for the evaluation of a number of later systems (and later versions of systems that participated in the QGSTEC'10), e.g., Agarwal et al. (2011), Aldabe et al. (2011), Aldabe, Gonzalez-Dios, Lopez-Gazpio, Madrazo, and

Maritxalar (2013), Olney et al. (2012) – a modified version, and Yao, Bouma, and Zhang (2012). Yet the scheme has some problems: The definitions of some scores are vague and leave a lot of room for interpretation, for example, the first three relevance scores distinguish between questions that are "completely relevant to the input sentence", questions that relate "mostly to the input sentence" and questions that are "related only slightly to the input sentence" (Rus et al., 2010, p. 53), the scores for syntactic correctness and fluency distinguish between questions that do "not read as fluently as [the authors] would like", questions with "some grammatical errors" and questions that are "grammatically unacceptable" (Rus et al., 2010, p. 54). It is difficult to draw a line between these scores, which is especially problematic if the annotator judging the quality of the questions is the author of the QG system. Another problem is the implicit independence assumption between the different criteria. The idea behind the different criteria is to evaluate how systems manage to cope with different challenges of question generation, but I think some basic criteria, especially grammaticality and acceptability, should be considered as prerequisite for evaluating other more subtle criteria like ambiguity and variety. A last problematic point is the computation of averages from ordinal scale data. This may seem a bit pedantic, but the use of this arithmetic operation implies that a grammatically correct but less fluent question (rank 2) is exactly twice as bad as a grammatical and idiomatic question (rank 1), two-thirds as bad as a question with some grammatical errors (rank 3) and only half as bad as a grammatically unacceptable question (rank 4). Apart from the scale-related problem, the arithmetic mean also hides potentially interesting properties of the distribution. A system that consistently generates grammatically correct but unidiomatic questions will score higher than a system that generates both idiomatic and ungrammatical questions half of the time, although the output of the second system might be easily fixed with an additional ranking component, while the first system might suffer from deeper problems that are difficult to solve.

Another evaluation scheme is proposed by Heilman (2011)[9]. He uses a single scale with scores from one to five, where higher scores are better than lower scores. The question rating is performed by three human raters. Ungrammatical or unidiomatic questions and questions that imply information incompatible with the text receive one of the two lowest scores. A question that receives the highest score is not only linguistically correct and specific, it is also supposed to be "as good as one that a teacher might write" (Heilman, 2011, p. 87). The scores in the middle are again somewhat vague – a question is 'acceptable' if it "does not have any problems", 'borderline' if it "might have a problem" and 'unacceptable' if it "definitely has a minor problem" (Heilman, 2011, p. 87).

[9]He also performs an extrinsic evaluation in form of a user study in an educational context.

A general problem that holds for evaluations of many text-based QG systems, independent of the rating scheme, is connected with the selection of the input. Since question rating has not been automated yet, it has to be done manually. This means that usually only questions generated from a relatively small set of sentences can be rated. These small sets are taken from a variety of different sources (with sentences from different domains and of different linguistic complexity). Systems that focus on certain linguistic phenomena additionally need to ensure that these phenomena occur in the evaluation data in sufficient frequency. Thus, they probably cannot randomly select the data from their chosen source (often it is not clearly stated whether the data were chosen randomly or following some other strategy).

All this means that we should not compare the results of quantitative analyses across different systems, except for those that were evaluated on the same data set and according to the same rating scheme, as in the QGSTEC'10.

2.3. Question Taxonomy

From a shallow linguistic point of view, questions can be classified according to their question phrase (as in the QGSTEC'10). However, the question phrase often neither reflects the computational complexity involved in generating a question nor the cognitive effort that is necessary to answer a question. Therefore, different taxonomies originating from the domains of psychology and education have been used to classify questions, especially in the evaluation of QG systems for educational applications[10]. The two most influential taxonomies are Bloom's (1956) taxonomy of educational objectives for the cognitive domain, which is not restricted to questions, and the educational question taxonomy by Graesser and Person (1994). A more recent taxonomy for tutoring questions which builds upon both Bloom (1956) and Graesser and Person (1994) is outlined by Nielsen, Buckingham, Knoll, Marsh, and Palen (2008). In the following, I will briefly discuss whether the categories from Graesser and Person's (1994) taxonomy are also applicable to the questions generated by the application-neutral QG system of this thesis.

Graesser and Person (1994, pp. 108–115) distinguish questions of different quality according to three dimensions: the *content of the requested information*, the underlying *question-generation mechanism* and the *degree of specification*. Definitions for the categories of the first dimension can be found in Table 2.3. The table lists 16 categories for syntactic questions and divides them into two groups according to the expected length of the answer (short or long); two additional categories are reserved for non-syntactic questions. The categories may not only be grouped by the length of the expected answer

[10] As we have seen above, most QG systems were developed for educational purposes.

Question category	**Abstract specification**
Short answer	
Verification	Is a fact true? Did an event occur?
Disjunctive	Is X or Y the case? Is X, Y, or Z the case?
Concept completion	Who? What? What is the referent of a noun argument slot?
Feature specification	What qualitative attributes does entity X have?
Quantification	What is the value of a quantitative variable? How many?
Long answer	
Definition	What does X mean?
Example	What is an example label or instance of the category?
Comparison	How is X similar to Y?
Interpretation	What concept or claim can be inferred from a static or active pattern of data?
Causal antecedent	What state or event causally led to an event or state?
Causal consequence	What are the consequences of an event or state?
Goal orientation	What are the motives or goals behind an agent's action?
Instrumental/procedural	What instrument or plan allows an agent to accomplish a goal?
Enablement	What object or resource allows an agent to perform an action?
Expectational	Why did some expected event not occur?
Judgmental	What value does the answer place on an idea or advice?
Assertion	The speaker makes a statement indicating he lacks knowledge or does not understand an idea.
Request/Directive	The speaker wants the listener to perform an action.

Table 2.3.: Question quality according to the content of the requested information, first proposed by Graesser et al. (1992), adapted from Graesser and Person (1994).

but also according to cognitive complexity: Graesser and Person (1994) distinguish "*deep-reasoning questions*, which elicit patterns of reasoning in logical, causal or goal-oriented systems" (p. 112), viz. questions from the category "antecedent, consequence, goal-orientation, instrumental/procedural, enablement, and expectational" (ibid.) from other questions. According them, "deep-reasoning questions are highly correlated with the deeper levels of cognition in Bloom's taxonomy of educational objectives (Graesser & Person, 1994, p. 112). However, in the context of text-based question generation, this categorization is misleading. The system developed in this thesis is not able to perform any complex reasoning, nevertheless, it is able to generate questions from all of the above-mentioned 'deep-reasoning categories', given that the information is explicitly encoded, for example, in an embedded clause:

(3) a. Die Bewohner flohen, weil das Haus in Flammen stand.
b. Warum flohen die Bewohner?

Answering question (3-b) after reading a text that contains sentence (3-a) does not require deep reasoning but only understanding and remembering sentence (3-a). Thus, according to Bloom's taxonomy, question (3-b) belongs to one of the lowest levels of abstraction (*knowledge*, or *comprehension* in the context of second-language acquisition), as do all the other questions generated by the system. Taxonomies like the one by Graesser and Person (1994) or Nielsen et al. (2008) that only consider the questions itself cannot assess the computational or cognitive complexity associated with a question[11]. To do this, we need to investigate the relation between textual input and generated questions as well as the relation between questions and expected answers.

The second dimension of question quality is concerned with different motivations behind questions. There are four categories: information-seeking questions, questions for managing common ground, questions for the coordination of social actions and conversational-control questions. The questions generated in this thesis probably would fall under the second category, as the answers are given in the text, but the categories are very much tailored towards tutorial dialogues and may not always be useful in other contexts (for instance, in the context of a simple reading comprehension system, the notion of *common ground* is a bit odd).

The degree of specification describes how specific or vague a question is. Since the questions generated by the system are supposed to be answerable after reading the whole input text, we should aim at a high degree of specification. Context-dependent expressions should be resolved or avoided.

2.4. General Challenges

The general challenges of question generation have already been discussed by a number of authors, usually in the context of English QG systems. The most comprehensive overview of challenges for factual question generation from a computational and linguistic point of view can be found in Heilman (2011, pp. 24–43). This section summarizes his overview and adds some critical remarks.

Heilman distinguishes challenges on three linguistic levels: lexical challenges, syntactic challenges and discourse challenges[12]. According to him, on the lexical level, a factual

[11] Nielsen et al. (2008) are aware of this problem. They point out that while the progression of their taxonomy "is consistent with the technical challenges involved, [they] believe all of the question types in the primary taxonomy can, under restricted conditions, be generated based on today's technologies" (Discussion, para. 4).

[12] A fourth class of "other challenges related to the use of QG tools in classrooms" (Heilman, 2011, p. 24) is ignored here, since the system of this thesis is application-neutral.

QG system first needs to find one or more appropriate question phrases for a given answer phrase. On a closer look, the process of generating question phrases actually does not operate on a purely lexical level. Often, the problem of determining certain morpho-syntactic and semantic features of a phrase can be reduced to a lexical problem: We first find the semantic and/or syntactic head of a phrase and then determine the relevant features on the lexical level. However, this is not always sufficient, sometimes we need to analyze the compositional semantics of a phrase or the semantic roles assigned by the verb. For example, to distinguish certain temporal NP subgroups, we need to take into account several features of the phrase, not just of the head, see section 4.2.1. To avoid repeating all lexical material from the source sentence, a QG system ideally should introduce some lexical variation. This thesis does not tackle this challenge, although it might be interesting to explore for this purpose the lexical-semantic net that is already used by the system to disambiguate and semantically classify words. Another problem that Heilman locates on the lexical level are expressions with non-compositional semantics. Again, the system developed here does not try to solve this issue and we will see in section 5.2.3 that this leads to some low-quality questions.

On the syntactic level, we need a constituency parser to identify the spans of potential answer phrases and their hierarchical position (to prevent the violation of syntactic movement restrictions). The performance of the whole system very much depends on the quality of the constituency parses, as we will see in the evaluation in section 5. A special problem are syntactically complex sentences: On the one hand, the performance of the parser is expected to drop with the length of the input, on the other hand, it might not always be possible to generate questions simply by substituting an answer phrase with a question phrase and moving it into the prefield due to movement constraints or the complexity of the resulting question. Sometimes sentences do not even state certain information explicitly but merely accommodate presuppositions (which may contain question-worthy information). To solve this problem, Heilman developed an algorithm that extracts a set of entailed sentences via syntactic simplifications and the exploitation of presupposition triggers. These entailed sentences are used as input to the system in addition to the original sentence. Developing such an extraction or text simplification module for German given the available automatic annotation would be worth another thesis[13], thus I leave this task for future work and focus on generating questions for sentences of 'average' syntactic complexity[14].

The last level is that of discourse-related problems. Heilman (2011) identifies two

[13] Already splitting simple VP coordinations involves dealing with elliptic structures that often go hand in hand with incorrect constituency parses. Removing non-restrictive relative clauses in German is a non-trivial task, since there are no orthographic markers which could be used to distinguish non-restrictive from restrictive relative clauses.

[14] A sentence extraction module may be added later, once the system works for less complex sentences.

categories of discourse challenges: "vagueness of information taken out of context" (p. 35) and "problems that result from implicit discourse relations and computers' lack of world knowledge" (ibid.). Since the second category of problems is not addressed in this thesis, I will only briefly discuss the first category. If we take sentences out of their discourse context, vagueness may result from different phenomena, such as different kinds of endophoric reference (introduced by pro-forms or demonstrative noun phrases) or the absence of information that was mentioned in the immediately preceding context of the sentence (in such cases, temporal or local adjuncts may be left out, fully specified phrases may be replaced by semantically underspecified phrases). A first step in preventing vague questions is the resolution of pronominal coreferences (see sections 3.7 and 4.4), but other pro-forms, such as pronominal adverbs and semantically underspecified noun phrases and predicates, still pose problems (see section 5.2.3) that are difficult to solve given the currently available NLP tools.

2.5. Challenges Specific to German

The German language poses a number of special challenges for question generation. Often we have to deal with a combination of language-specific complexity and restricted computational linguistic means. Thus, it does not make sense to separate the theoretical linguistic discussion from practical considerations about available linguistic features and implementation details. This section is only supposed to provide the reader with a first impression of the complexities involved in building a German QG system. In-depth discussions of linguistic and computational problems can be found in the respective sections of the chapters on automatic linguistic annotation and question generation.

The first German-specific problem is the complex mapping from answer phrases to question phrases. Consider, for example, the selection of different question phrases required by prepositional phrases headed by *an* in (4).

(4) a. Sie denken *an Peter*. – an wen
b. Sie denken *an das Meer*. – woran
c. Sie gehen *an die Bar*. – wohin
d. Sie stehen *an der Bar*. – wo
e. *Am Schiedsrichter* gibt es nichts zu kritisieren. – an wem
f. *An jedem zweiten Sonntag* ist Markttag. – wann

The first example seems similar to its English equivalent[15]: We simply generate the

[15]Depending on the variety of English, formal register and grade of prescriptiveness, we may find the parallel version *Of whom are you thinking?* (case-marked interrogative pronoun, pied-piped preposition), *Whom are you thinking of?* (case-marked interrogative pronoun, stranded preposition) or *Who are you thinking of?* (uninflected *who*, stranded preposition).

appropriate question word for the NP and prepend the preposition. However, we first need to identify the correct case to account for the difference between (4-a) and (4-e). From (4-b), we can tell that distinguishing the case is not sufficient. Apparently, noun phrases referring to non-persons follow a different pattern: The preposition is appended to *wo* with some kind of linking element. By looking at examples (4-b), (4-d) and (4-f), we must conclude that even this new generalization does not capture the full picture yet. Prepositional phrases that function as modifiers seem to follow more complex rules[16] and the preposition is not always part of the question phrase.

Pronouns pose a twofold problem in question generation for German text: Like in English, if they appear out of context, it is unclear what they refer to. So, any pronoun whose antecedent is not in the question itself, should be resolved. Unlike in English, correct question phrases cannot be generated from unresolved (third person) personal pronouns. In (5), *ihn*, a masculine third person singular personal pronoun, refers to Maria's keys. Assuming that *ihn* always refers to a male person results in the ill-formed question in (5-a).

(5) Maria sucht *ihren Schlüssel*. Zuletzt hat sie *ihn* auf dem Tisch liegen sehen.

a. **Wen* hat Maria zuletzt auf dem Tisch liegen sehen?

b. *Was* hat Maria zuletzt auf dem Tisch liegen sehen?

The problem is that third person personal pronouns in German agree with the gender of their antecedent, but do not encode information on the animacy of their referent, whereas substituting interrogative pronouns are sensitive to animacy, not gender. Only the nominal head of its antecedent can tell us whether a pronoun refers to a person (*wer*, *wen*, *wem*) or not (*was*). Thus, resolving coreferences in a German QG system is even more crucial than in an English one.

Once we have a question phrase, we need to transform the declarative sentence into a question. Therefore, we need a word order model ensuring well-formed output. The difficulty lies in finding a word order that works independent of the immediate context of the source sentence. Many grammatical word orders depend on information-structural properties of the context and thus have to be avoided. A detailed investigation of this problem is carried out in section 4.3.

If we ask for plural subjects, we need to fix the number agreement on the verb (6-b). If we replace a pronoun by its antecedent, we need to adjust the antecedent's case marking to the case of the pronoun (6-d).

(6) a. Im Winter fliegen die Urlauber in den Süden.

b. Wer *fliegt* im Winter in den Süden?

[16] A similar observation holds for noun phrases, see section 4.2.1.

c. Maria beobachtet den Polizisten auf der anderen Straßenseite. Er nimmt einen Drogendealer fest.
d. Wen nimmt *der Polizist* auf der anderen Straßenseite fest?

To perform these operations, the system needs an inflection model covering both conjugation and declension.

3. Automatic Linguistic Annotation

The assumed input to the question generation system is unstructured German text. In order to identify potential answer phrases and transform declarative sentences into questions, the data needs to be annotated with information of different linguistic levels.

3.1. Basic Linguistic Units

In a first step, basic linguistic units need to be identified: The system uses the OpenNLP maximum entropy sentence detector and tokenizer[1] with the pretained models[2] to split the text into sentences and tokens. To preserve whitespace information, the API method that returns spans of the units in the original string is used. Since both models were trained on the TIGER corpus, which encodes opening quotation marks as ``` `` ``` and closing quotation marks as `''`, they cannot handle regular quotation marks (neither straight nor typographical). Thus, opening quotation marks are replaced by two back ticks, closing quotation marks by two apostrophes. For straight quotation marks, opening symbols are identified with the regular expression `\"(?!\\s|\\z|[.!?,;])`, all remaining straight quotation marks are replaced by two apostrophes by default. This possibly leads to some inaccuracies, since the same symbol in different contexts may represent inches, seconds or arc degrees, but these cases are probably rare and the effects only cosmetic.

3.2. Part-of-speech Tags

Part-of-speech tags provide useful linguistic information in their own right, but they are particularly important for syntactic parsing. Since the system's performance heavily depends on the accuracy of constituent and dependency parses, it is not the quality of the POS tags itself, but the quality of the parses generated with these tags, on which I based my choice of the POS tagger. I manually inspected sample parses for three

[1] All OpenNLP software is licensed under the Apache License, Version 2.0, and can be found at `https://opennlp.apache.org/`. The usage of the tools is described in a manual: `https://opennlp.apache.org/documentation/1.5.3/manual/opennlp.html`.

[2] Pretrained OpenNLP models for different languages are available at `http://opennlp.sourceforge.net/models-1.5/`.

```
(ROOT
  (NUR
    (S (PPER Er) (VVFIN ging)
      (PP (APPR nach) (NN Hause))
      ($, ,)
      (S (KOUS weil) (PPER er)
        (NP (ART die) (NN Katze))
        (VVFIN füttern)))
    (VVFIN musste) ($. .)))
```

Figure 3.1.: Stanford parse based on tags from the Stanford tagger.

taggers (all three produce tags that adhere to the *Stuttgart-Tübingen Tag Set* (STTS)[3], as required by parser models trained on NEGRA or TIGER) and found that the results differ considerably. In the following, I illustrate two observed patterns with Stanford parses (see section 3.3) of the example sentence in (1).

(1) Er ging nach Hause, weil er die Katze füttern musste.

The Stanford maximum entropy tagger (Toutanova & Manning, 2000), trained on the same data as the German Stanford parser, sometimes falsely tags infinitive verbs as finite. This causes the parser to assume unconnected structures, resulting in a special *non-unary root* (NUR) category under the root node, see the example parse in figure 3.1. A parse with this kind of structure is very difficult to handle and without some (non-trivial) heuristic post-processing the system's output is likely to be ill-formed.

The TreeTagger (Schmid, 1994, 1995) correctly tags *füttern* as a finite modal verb (VMFIN), but apparently the Stanford parser does not handle this very well (at least in this and similar examples; maybe due to a sparsity of modal verbs in the training data). Instead of a sentence with a simple embedded clause, the parse in figure 3.2 contains two coordinated sentences. Based on this kind of parse, the system may fail in two different ways: 1) If coordinated sentences are extracted in a preprocessing step, the function of the embedded clause in the matrix sentence will not be recognized. The system will not be able to ask, why he went home, and it will have difficulties interpreting the second sentence, as the embedding sentence may affect its meaning, for example, if the embedded clause is in the scope of negation, the meaning might be reversed, if the embedded clause is the complement of a modal verb or a verbum dicendi, it might not even have a truth value and should not be treated as a declarative sentence. 2) If simple sentences are not extracted from coordinations, no questions will be generated because

[3]Schiller et al. (1999) describe the tags in great detail. A table with the original tagset and the TIGER modifications can also be found in Appendix A.1.

```
(ROOT
  (CS
    (S (PPER Er) (VVFIN ging)
      (PP (APPR nach) (NN Hause)))
    ($, ,)
    (S (KOUS weil) (PPER er)
      (VP
        (NP (ART die) (NN Katze))
        (VVINF füttern))
      (VMFIN musste))
```

Figure 3.2.: Stanford parse based on tags from the TreeTagger.

coordinations are islands for extraction.

The Mate tagger[4] identifies *füttern* more coarsely as finite verb (VVFIN). The syntactic parse is as expected, a matrix sentence with a simple embedded clause. Since the Mate tagger yielded the best results for my test set and the system also uses the Mate dependency parser, I decided to stick with this alternative.

3.3. Constituency Parse

Constituency parses are crucial for identifying potential answer phrases. As mentioned above, errors in the constituency parse are very likely to result in an overall poor performance of the system. For example, if potential answer phrases do not form a constituent in the parse, the system cannot ask for them; if the parser gets the constituent labels wrong, a wrong question phrase will be generated; if certain dominance relations are not recognized by the parser, the generated questions might violate syntactic movement restrictions; and if constituents span over more tokens than they should, moving these constituents is certainly going to result in ungrammatical structures.

I tested both Stanford's lexicalized PCFG parser[5] (D. Klein & Manning, 2003; Rafferty & Manning, 2008) and the Berkeley parser[6], both with their distributed German models pretrained on the NEGRA treebank (Stanford) and the TIGER treebank (Berkeley). Models trained on the NEGRA or TIGER treebank have two advantages over models trained on TüBa-D/Z: 1) Their grammar formalism, which builds on functor-argument relations, facilitates identifying answer phrases and manipulating syntactic structures.

[4]The Mate tools for natural language analysis are available under *GNU General Public License*, version 3, at `https://code.google.com/p/mate-tools/`.

[5]All Stanford NLP software is available under *GNU General Public License*, version 3, at `http://nlp.stanford.edu/software/index.shtml`. The source code is on GitHub: `https://github.com/stanfordnlp/CoreNLP`.

[6]The Berkeley parser is available at `https://github.com/slavpetrov/berkeleyparser`.

For example, when moving the prefield constituent back into its base position (generating a verb-first sentence with unmarked word order before the insertion of the question phrase, see section 4.3), we only have to consider target positions between the immediate children of the topmost S node. 2) There are models trained on NEGRA or TIGER data available for the most recent versions of both parsers, this is not the case for TüBa-D/Z.

In contrast to the taggers tested above, there was no clear winner here. Each parser showed different errors that pose different problems for the question generation system. For all experiments in this thesis, I use the Stanford parser because the Berkeley parser sometimes seems to confuse constituency labels, but all methods are designed such that they may operate on the output of both parsers[7]. To enhance the resulting parses, some heuristic transformations are applied: If a noun or a noun phrase is followed by a comma and a relative clause, then these elements are grouped under a new complex noun phrase. If a prepositional phrase is followed by a comma and a relative clause, the comma and the clause (and the comma after the clause, if there is one) are moved under the prepositional phrase.

3.4. Morphologically Rich Tags and Lemmas

To get detailed morphological features, the tokens are further annotated with morphologically rich tags[8] by the RFTagger[9] (Schmid & Laws, 2008). These tags replace the simple tags (i.e. the values of the preterminal nodes) in the parse tree, allowing for more detailed *Tregex* queries later on. Based on the token and its morphologically rich tag, the corresponding lemma is obtained from the lemmatizer that comes with the RFTagger. There is a variety of tools available both for morphological analysis[10] and lemmatization[11]. The RFTagger and its lemmatizer are used because the inflection model (described in section 4.5.1) is based on the lemma lexicon of this lemmatizer. Using another tagger or lemmatizer would increase the probability of inflection failures, especially since the RFTagger lemmas for nouns, pronouns, determiners and adjectives are feminine, while others, for example, the Mate lemmatizer, return masculine forms.

[7] The Berkeley parser produces an additional PSEUDO node below the root node, which contains both the sentence and sentence final punctuation, whereas the Stanford parser puts the S node directly under the root and sentence final punctuation below the S node.

[8] The tags adhere to the big STTS (Schiller et al., 1999). The morphological features are summarized in Appendix A.2.

[9] The tagger is available at `http://www.cis.uni-muenchen.de/~schmid/tools/RFTagger/`. I used it via the Java interface by Ramon Ziai and Niels Ott at `http://sifnos.sfs.uni-tuebingen.de/resource/A4/rftj/`.

[10] See, for example, the morphology options in ParZu's pipeline in section 3.7, footnote 19.

[11] German lemmatizers are also included in the Mate tools and the TreeTagger distribution.

3.5. Dependency Information

The input sentences are also parsed with the graph-based Mate dependency parser (Bohnet, 2010) using a model trained on the dependency conversion of the TIGER data described in Seeker and Kuhn (2012).

3.5.1. Lexical Heads

One motivation for the use of a dependency parser was a problem that arises due to the nature of the syntactic annotations in NEGRA and TIGER. Nominal and prepositional phrases have a relatively flat internal structure and thus finding the lexical head based solely on the constituency parse can be hard. An implementation of a head finder that adapts the head rules in Collins (1999, Appendix A) to the NEGRA corpus is included in the Stanford CoreNLP tools. However, these rules do not always find the expected head and are not of much help in the case of NEGRA's flat PPs: The head finder will return the preposition as head of the PP, but cannot find the nominal head that is governed by the preposition if there is no NP.

Instead of using heuristics based only on the constituency parse, I decided to inform the head choice by additional dependency information. The lexical element with the largest number of dependents is assumed to be head of the phrase. The notion of dependency in this case is a transitive one: If a is a dependent of b and b is a dependent of c, then a is also a dependent of c. Another way to view this solution without a transitive notion of dependency, is in terms of a *PageRank*-like algorithm: The score of a lexical element depends on the number of its immediate dependents and the scores of these dependents. To find the head of an implicit NP under a PP, the search space is restricted to nominal elements. Let me illustrate the dependency-based head finder with a simple example: The constituent parser systematically fails to recognize prenominal genitive NPs, which results in false multi-word proper nouns with a flat internal structure, e.g., *[$_{MPN}$ [$_{N}$ Peter] [$_{N}$ Pans] [$_{N}$ Wagen]]* for the subject of sentence (2).

(2) Peter Pans Wagen hat eine Panne.

Figure 3.3 contains the dependency parse for (2) with the number of transitive dependents for each word[12]. Since *Wagen* has more dependents than both *Peter* and *Pans*, it will be annotated as the head of the phrase.

[12]Punctuation is ignored, which is why *hat* has only five dependents and *Panne* only one.

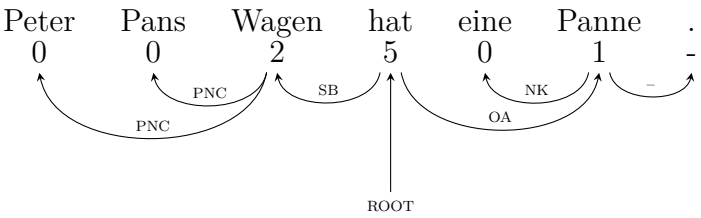

Figure 3.3.: Dependency parse with number of transitive dependents for each word.

3.5.2. Grammatical Functions

It is possible to obtain grammatical functions from a constituent parser with a model trained on a modified treebank, where edge labels are concatenated to dependent phrasal categories. However, Rafferty and Manning (2008) found that doing so leads to a considerable drop in performance, especially for TIGER (more than 15%). Further tests revealed that "adding grammatical functions is not only problematic due to increased categorization but because of sparseness" (Rafferty & Manning, 2008, p. 43), which means that training with grammatical functions even diminishes the parser's performance on basic categories.

These problems can be avoided with a dependency parser[13]. The dependency parser annotates lexical elements with the label of their incoming edge (i.e. the relation between the lexical element and its head)[14]. To obtain grammatical functions of non-lexical constituents, I annotate them with the edge labels of their lexical heads. This is useful for a number of different tasks, for example, identifying appropriate question phrases given an answer phrase or linearly reordering sentence-level constituents.

3.6. Semantic Classes and Groups

The system needs to guess appropriate question words based on identified answer phrases. For this task it can use morphological information and edge labels from the dependency parse (e.g., to choose between *wer*, *wen* or *wem*), but to decide upon the correct question word lexeme, often semantic information about head nouns in NPs and PPs is necessary. Heilman (2011) obtained this information from the *Supersense Tagger* (Ciaramita & Altun, 2006), but since this tool is not available for German, I had to find a different solution: In a first step all named entities are identified and classified by the Stanford named entity recognizer (NER) using the German model trained by Sebastian Padó (see Table 3.1 for entity types and examples[15], Faruqui and Padó (2010)). The named

[13]For a performance evaluation of the Mate dependency parser, see Bohnet (2010, p. 96).
[14]See Appendix A.4 for an overview of all NEGRA and TIGER edge labels.
[15]Actually, *Siemens* is not recognized as ORG but belongs to this type according to the annotation guidelines (`http://www.cnts.ua.ac.be/conll2003/ner/annotation.txt`).

Entity Type	Examples
person (PER)	Maria, Lehmann
location (LOC)	Leipzig, Deutschland
organisation (ORG)	Siemens, SPD
miscellaneous (MISC)	Deutsch, Kommunismus

Table 3.1.: Entity types recognized by Faruqui and Padó (2010).

entity recognizer often fails to recognize first names. I fixed this issue with two lists of first names (female and male) borrowed from the coreference tool CorZu (Klenner & Tuggener, 2011, see section 3.7).

In a second step, all remaining tokens with nominal and verbal[16] tags are classified with one of GermaNet's (Hamp & Feldweg, 1997; Henrich & Hinrichs, 2010) semantic field labels. To find the correct semantic field for a word, polysemous and homographic words first need to be disambiguated. This is done via the Lesk algorithm in GermaNet's *Relatedness* library as described in Henrich and Hinrichs (2012, p. 578): For each word in a sentence, a set of *synsets* is retrieved from GermaNet. A synset is a set of synonymous lexical units, or roughly speaking, GermaNet's equivalent of a word sense. For each synset $t \in T$ belonging to the target word, a Lesk relatedness value $l(t, c)$ is computed for each context synset $c \in C$ (retrieved for the remaining words of the sentence). The computed values are added for each target synset and the synset with the maximal value is chosen as the most probable word sense d:

$$d = \arg\max_{t \in T} \sum_{t,\, c \in C} l(t, c) \tag{3.1}$$

If all Lesk values are zero, the first synset is chosen by default, in case of ties, the synset whose relatedness value was computed first. From the chosen synset, the word class can be accessed directly.

Based on the semantic classes from the named entity recognizer and GermaNet's semantic fields, I define the nominal semantic groups shown in Table 3.2. A semantic group is a set of semantic classes that share certain features relevant for question generation. The groups *Person*, *Time* and *Location* are self-explanatory; *Object* contains nouns that have a spatial extension (excluding those from the *Person* group) and is complementary to *Abstract_Entity*; *Group* contains all kinds of organizations, e.g., companies or political movements. The rules presented in section 4.2 may refer directly to semantic classes like *Kommunikation*, *Menge* and *PER* or to semantic groups.

[16]Sometimes, we need the category of the 'verbal semantic head' of a sentence, see section 4.2.1.

Semantic group	Set of semantic classes
Person	Mensch, Person, PER
Group	Gruppe, ORG
Time	Zeit
Location	Ort, LOC
Object	Artefakt, Koerper, Nahrung, natGegenstand, natPhaenomen, Pflanze, Substanz
Abstract_Entity	Attribut, Form, Gefuehl, Kognition, Kommunikation, Motiv, Relation, Tops, MISC

Table 3.2.: Sets of semantic classes that make up semantic groups.

3.7. Coreference

Coreference resolution is done by CorZu, a rule-based system developed at the University of Zurich, which achieved the best results in the SemEval coreference task 2010 (Klenner & Tuggener, 2011) and still seems to fare quite well compared to HotCoref DE, a German coreference resolution system recently developed at the University of Stuttgart (Rösiger & Riester, 2015, see page 86 for a comparison with CorZu). As input, Corzu needs tokens, basic tags, STTS-tags, morphological features and dependency information (edge labels and heads). In principle, all these data could be collected from the output of the previously described tools, but this would involve converting morphological features and edge labels to another format. To avoid this error-prone conversion and to make sure that the tool receives the input it was developed for (possibly also with the same systematic mistakes), I added ParZu, the dependency parser of the University of Zurich, to the pipeline. ParZu itself contains a full pipeline that splits a text into sentences and tokens[17], assigns basic tags and STTS-tag[18], analyzes the morphology of each word with Zmorge[19], the Zurich morphology analyzer for German (Sennrich & Kunz, 2014), and finally performs dependency parsing. I skip ParZu's sentencer, tokenizer and tagger, and provide it with sentences, tokens and STTS-tags from the tools described above.

CorZu's output is in the CoNLL-2011 format[20], in which each token (together with its annotation) is on a separate line and sentences are separated by an empty line. The last column contains the coreference annotation. Coreferring elements are indicated by token spans annotated with the same coreference index. A span can be embedded in another span, e.g., in the first of the example sentences that come with the system,

[17]For this, ParZu uses a modified version of the NLTK Punkt sentence tokenizer (Bird, Klein, & Loper, 2009).

[18]By default they use their own Clevertagger (Sennrich, Volk, & Schneider, 2013). The distribution also contains a python wrapper for the TreeTagger.

[19]Actually, there are also options for SMOR (Schmid, Fitschen, & Heid, 2004) and GERTWOL (Koskeniemmi & Haapalainen, 1994), but I chose Zmorge because of its Creative Commons license.

[20]A detailed description can be found here: `http://conll.cemantix.org/2011/data.html`.

there is a noun phrase with restrictive relative clause and both the whole complex noun phrase and the relative pronoun inside that phrase are annotated as spans with the same coreference index:

(3) [1 ein Handy , [1 das] auf die CDU zugelassen sei]

In these cases, only the maximal span is considered as potential antecedent. The system reads in the CoNLL file and annotates each constituent in the syntactic parse that matches a coreference span with the corresponding coreference index. Once the file has been processed, we have several coreference chains consisting of subtrees from the constituency parses. A copy of the subtree of the first element in each coreference chain is then added to all its subsequent elements, assuming that it always provides the referent. This is not true for cataphoras, which do not depend on an antecedent, but a postcedent. However, CorZu treats them as regular anaphors, so cataphoras will cause problems anyway.

3.8. Data Structures

All annotations are stored in Stanford's `Tree` class, or more precisely in `TreeGraphNode`s extending the abstract `Tree` class. Each `TreeGraphNode` holds a `CoreLabel`, a map from keys to values, which allows adding annotation to any constituent in the tree, and a pointer to its parent node. There are predefined keys[21] available for different types of annotations, e.g., tokens, lemmas and tags; new keys can be defined as needed, making it possible to annotate constituents with data of any primitive or composite type, for example, pronouns with a subtree representing their antecedent.

The main advantages of using Stanford's classes over custom classes are the following: 1) They are flexible, allowing annotations of any type at any level of the tree, as already mentioned above. 2) `CoreLabel`s are backed by memory-optimized data structures, which can be important because each tree contains a lot of them. 3) Stanford's classes are widely used, they come with extensive documentation and there is some support via GitHub and a mailing list. This makes the code more readable and re-usable and less error-prone. 4) Many Stanford tools take lists of `CoreLabel`s as input, for example, the tagger, the parser and the named entity recognizer mentioned above. 5) Stanford `Tree`s can be searched with `Tregex` ('tree regular expressions') and transformed with `Tsurgeon`[22], a tree transformation language (Levy & Andrew, 2006).

[21] A key is a class that implements the `CoreAnnotation` interface.

[22] I use the `Tree` API instead of `Tsurgeon` because it is often more efficient and convenient (especially for nodes with parent pointers), but `Tsurgeon` is always an option.

4. Question Generation

This chapter describes the core of the question generation system. Once the input text is fully annotated, the first step is to identify potential answer phrases (section 4.1). Based on the identified answer phrases, the system generates appropriate question phrases[1] (section 4.2). Since the answer phrase need not always occupy the prefield, a linearization component is developed, which moves the prefield constituent into the right position if necessary (section 4.3). After the final order of constituents is determined, pronominal coreferences are resolved based on the annotation from the previous chapter (section 4.4). Finally, the inflection of finite verbs, antecedents and possessive pronouns is adjusted (section 4.5) and some cosmetic post-processing steps are performed (section 4.6).

4.1. Identifying Potential Answer Phrases

To identify answer phrases, Heilman (2011) uses 18 Tregex patterns to mark unmovable constituents[2]. Anything that is not marked is considered as potential answer phrase. I adapted this approach to the syntactic structures in the NEGRA treebank.

First, I define a set of syntactic categories that qualify as potential answer phrases ('markables') regardless of hierarchical constraints, see the Tregex macro in (1-a). This set contains nominal phrases (NP), prepositional phrases (PP), clausal constituents (S) and coordinations of any of these (CNP, CPP, CS). Additionally, the set contains personal pronouns (PRO-Pers), multi-word proper nouns (MPN, e.g., *Tobias Kolditz*) and nouns (N, because not all nouns are under NPs in NEGRA). For convencience, I define a second macro without S and CS, see (1-b).

(1) Tregex macros

a. `ALL_MARKABLES /^(PP|CPP|CNP|S|CS|N|MPN|PRO-Pers)/`

b. `NON-S_MARKABLES /^(PP|CPP|CNP|N|MPN|PRO-Pers)/`

Due to syntactic movement restrictions, the categories defined in (1-a) sometimes cannot be asked for. In such cases, they are marked as 'unmovable' by the five Tregex patterns

[1] This can be a simple question word (e.g., *wer* or *wann*) or a more complex phrase (e.g., *für wen* or *welcher Schlüssel*).

[2] These patterns can be found in Appendix B.1.

	Tregex pattern
1	`ALL_MARKABLES=unmovable >> (@S >> @S)`
2	`ALL_MARKABLES=unmovable >> /^C/`
3	`/-PAR$/=unmovable`
4	`ALL_MARKABLES=unmovable >> NON-S_MARKABLES`
5	`ALL_MARKABLES=unmovable >> @UNMOVABLE`

Table 4.1.: Tregex patterns marking constituents as unmovable.

listed in Table 4.1.

The first pattern marks anything under a subordinate clause as unmovable (subjacency principle). This is too restrictive, but prevents a lot of ill-formed output given the current NLP tools (see section 4.1.3). The second pattern prohibits movements out of a coordinated phrase and is equivalent to Heilman's third rule. The third pattern marks parentheticals as unmovable. Patterns four and five are equivalent to Heilman's (2011) last two rules, which "are applied in order to propagate the constraints down the trees" (p. 181): Any descendant of an unmovable node and any descendant of a potential answer phrase is marked as unmovable.

But what happened to the other 13 rules? In the following, I will briefly discuss the three main reasons for the reduction in the number of rules.

4.1.1. Redundant Rules

Although rules 8, 9 and 10 in Heilman (2011, appendix A) are linguistically well motivated, they seem to be redundant. Rule 8 marks prepositional phrases dominated by a nominal phrase, with a preposition other than *of* or *about*. This rule is supposed to disallow (2-d), while still allowing for questions like (2-b).

(2) Examples for rule 8 (Heilman, 2011, p. 179)

a. John visited the capital of Alaska.
b. What did John visit the capital of?
c. John visited a city in Alaska.
d. *What did John visit a city in?

However, according to the rules that propagate the constraints down the trees, a question like (2-b) can never be generated and questions like (2-d) are ruled out anyway. There are two possible cases: Either the NP dominating the PP is a potential answer phrase and all its descendants are marked as unmovable because of rule 17, equivalent to pattern 4 above (this seems to be the case in (2-a)), or the NP is unmovable, in which case again all descendants are marked as unmovable according to rule 18, equivalent to pattern 5

above.

Rule 9 "is used to mark prepositional phrases that are nested within other prepositional phrases" (Heilman, 2011, p. 180). Again, there are two cases: The PP is either movable or not. In any case, it is an island for extraction according to rules 17 and 18.

Rule 10 "is used to mark prepositional phrases in subjects" (Heilman, 2011, p. 180), where the subject is defined as an NP that is a sister of a VP. The same argument as above applies here.

4.1.2. Restrictions of Non-Syntactic Nature

As Heilman (2011) states himself, rules 13 – 16 are only there to mark certain constituents that the system cannot handle. For example, rule 15 marks prepositional phrases whose preposition does not govern a noun phrase (e.g., *by tomorrow*, *after calling his girlfriend*). While my system shares most of the shortcomings that motivated these rules, I tried to separate system-inherent constraints from movement restrictions and consequently did not adopt these four rules.

4.1.3. Extractions from Finite Subordinate Clauses

At least seven rules[3] in Heilman (2011) restrict extractions from finite and non-finite subordinate clauses. My system can handle extractions from non-finite subordinate clauses (annotated as VPs in TIGER); questions like (3), however, currently cannot be generated because the first pattern in Table 4.1 prevents any extractions from finite subordinate clauses.

(3) Wen glaubt Peter, liebt Maria?

These extractions in German are subject to a number of, at first sight, very heterogeneous constraints, which need to be considered in order not to produce unacceptable or even ungrammatical questions. Modeling constraints on extractions from finite subordinate clauses is problematic mainly for two reasons: 1) Some of the linguistic properties referred to by these constraints cannot be (reliably) annotated by the tools described in section 3. 2) The constraints differ for speakers with different dialectal backgrounds. This section is supposed to give a first impression of the complexities involved in generating long-distance wh-movement questions in German[4] and serves as an excuse for not tackling the issue and possibly as a starting point for future work.

[3] Rules 1, 4, 5, 6, 7, 11 and 12.

[4] For a more comprehensive descriptive overview on the topic, I recommend Lühr (1988) and especially Andersson and Kvam (1984) with a lot of interesting observations and examples.

The first constraint is on the clause type: wh-words can only be extracted from object clauses, see (5-a) and (5-b) for unsuccessful extractions from an attributive and an adverbial clause.

(4) a. Er hatte die Hoffnung, dass er den Abschluss schaffen würde.
b. Er verließ die Uni, nachdem er seinen Abschluss erhalten hatte.

(5) a. *Was hatte er die Hoffnung, dass er schaffen würde?
b. *Was verließ er die Uni, nachdem er erhalten hatte?

The object clause may not start with an overt subordinating conjunction, with the only exception being *dass*, see the examples in (6). Although, for more complex examples, extractions from *dass*-clauses do not seem to work well, see example (12-a).

(6) a. ?Wen glaubt Peter, dass Maria liebt?
b. *Wen fragt Peter, ob Maria liebt?

The subordinate clause may not be in the scope of negation (7-a). Andersson and Kvam (1984, Appendix 1, example 34) cite the sentence in (7-c) as answer to the question in (7-b) from a conversation about cross-country skiing in Freiburg. This sentence violates the last two constraints and is unacceptable according to my intuitions. Andersson and Kvam (1984, p. 56) say that these constructions are less frequent, but acceptable if the extracted constituent is an adverbial prepositional phrase. The reason for the discrepancy between our judgements could be diatopic variation or diachronic change[5].

(7) a. *Wen glaubt Peter nicht, dass Maria wirklich liebt.
b. Könnte man eigentlich die Loipe nach Hinterzarten an einem Tag schaffen?
c. ?An einem Tag weiß ich nicht, ob man die Strecke schaffen würde.

Another constraint is on the verb type: Lühr (1988, p. 81) identifies three main groups that allow long-distance wh-movements: epistemic verbs (8-a), verbs expressing wish (8-b) and verba dicendi (8-c). The example in (8-d), borrowed from Gallmann (2015, p. 6), shows how the wrong verb type leads to an ungrammatical question.

(8) a. Wen glaubst du, hat er gesehen?
b. Was hoffst er, dass sie ihm schenken wird?
c. Wen sagt er, habe er gesehen.
d. *Wen bewirkte Otto, dass Anna auch einlädt?

[5] Andersson and Kvam (1984, pp. 104–107) discuss why these constructions are on the retreat since the first half of the 19th century.

The closest approximation to these three verb classes the system currently could get are GermaNet's semantic fields *Kognition* and *Kommunikation*, but, despite a big overlap, there are verbs belonging to one of the two semantic classes, which do not seem to allow long-distance wh-extraction, for example, *herausfinden* (9).

(9) ?Was fand Maria heraus, dass Peter gegessen hatte?

Any such verb will cause the system to generate a number of unacceptable questions (depending on which constituents the system is allowed to extract from the embedded clause, see below).

There seems to be a hierarchy of constituents with different grammatical functions that lead to varying degrees of acceptability when extracted from a finite subordinate clause: Accusative objects work well as potential answer phrases in clauses without overt complementizer (10-a). The other examples, although maybe still acceptable, to me sound less good, the next best maybe being the extracted prepositional object (11-b) and the extracted subject (10-b).

(10) Peter sagt, ein Mann habe seinem Sohn gestern vor der Schule Drogen verkauft.

a. Was sagt Peter, habe ein Mann gestern vor der Schule seinem Sohn verkauft?
b. ?Wer sagt Peter, habe seinem Sohn gestern vor der Schule Drogen verkauft?
c. ?Wo sagt Peter, habe gestern ein Mann seinem Sohn Drogen verkauft?
d. ?Wann sagt Peter, habe ein Mann seinem Sohn vor der Schule Drogen verkauft?

(11) a. Peter sagt, er könne sich auf Maria verlassen.
b. ?Auf wen sagt Peter, könne er sich verlassen?

The sentence in (12) is a reformulation of the one in (10) with the overt complementizer *dass* (triggering verb-last structure in the subordinate clause). To me now even question (12-a) seems unacceptable and question (12-b) completely ungrammatical. The reading where *wo* and *wann* in (12-c) and (12-d) refer to the embedded clause is difficult to get for me – introducing more material in the matrix clause (e.g., replacing *Peter* with *der Vater, der sich um seine Kinder sorgt*) or shifting the matrix verb to past tense in (12-d) makes it almost impossible. However, from personal communication I know that almost all these examples seem perfectly acceptable to some speakers from southern Germany.

(12) Peter sagt, dass ein Mann seinem Sohn gestern vor der Schule Drogen verkauft hat.

a. ??Was sagt Peter, dass ein Mann seinem Sohn gestern vor der Schule verkauft hat?

b. *Wer sagt Peter, dass seinem Sohn gestern vor der Schule Drogen verkauft hat?

c. ??Wo sagt Peter, dass ein Mann seinem Sohn gestern Drogen verkauft hat?

d. ??Wann sagt Peter, dass ein Mann seinem Sohn vor der Schule Drogen verkauft hat?

The RFTagger often confuses nominative and accusative case, which is also a problem for question phrase generation, but only if the answer phrase refers to a person (otherwise the case syncretism of *was* saves us). Intermediate tests have shown that mistakes in the annotation of grammatical case (and occasionally also grammatical function) are the most import reason for generating unacceptable long-distance wh-extraction questions.

So, we can only be sure to generate acceptable questions when the system is able to identify a constituent as an accusative object in a V2 complement clause of a verb from the three groups mentioned above that is not in the scope of negation. And while the output in this case is probably going to be acceptable, I doubt that it will be appropriate in all contexts. For example, if we want to generate reading comprehension questions for second language learners, questions like (10-a) are unnecessarily complex. We might rather want to generate questions like (13-a) or split the question in two separate parts (13-b).

(13) a. Was sagt Peter, wer seinem Sohn gestern vor der Schule Drogen verkauft hat?

b. Was sagt Peter? Wer hat seinem Sohn gestern vor der Schule Drogen verkauft?

The latter option has the advantage that we would not need to change the system, we would simply generate a question for the matrix clause and a separate question for the embedded clause.

4.2. Generating Question Phrases from Answer Phrases

For English, generating question phrases for a given answer phrase is not a particularly challenging task. For example, Heilman (2011, p. 64) defines six rules covering noun phrases as well as local and temporal prepositional phrases[6]. For the remaining prepositional phrases, the question phrase is generated for the noun phrase governed by the preposition and then moved to the beginning of the question, see question (14-a) with

[6]However, some problems (like different subgroups of temporal NPs) are not addressed by these rules, so maybe the generation of question phrases in English is not that easy after all.

stranded preposition (this is the question Heilman's system would generate) or question (14-b) with 'pied-piped' preposition.

(14) a. Where does John come from?
b. From where does John come?

For colloquial German, a similar strategy might be successful in certain cases, see the question in (15-a) asking for a prepositional phrase. However, the standard way of asking this question is (15-b).

(15) a. Von wo kommt John?
b. Woher kommt John?

While there are some regularities, many question phrases cannot be predicted from general rules. If there was a corpus of German texts where potential answer phrases were annotated with corresponding question phrases, this would be a classical use case for machine learning: One would define a set of features for each possible answer phrase type and let the machine learning algorithm figure out a mapping to question phrases, for example in the form of a decision tree. However, since there is no such corpus, I had to manually write rules for all possible question phrases. Each rule maps a set of answer-phrase features to a question phrase. Rules apply according to the principle of underspecification: A rule matches an answer phrases if its features form a subset of the features extracted from the answer phrase (*subset principle*); a rule applies, if it is the most specific rule that matches the answer phrase (*principle of specificity*). A rule with $n + 1$ features is more specific than a rule with n features; a hierarchy of specificity among different features decides ties.

The following three sections describe how the system generates question phrases for NPs, PPs and embedded clauses that were not marked as unmovable in the previous step. I do not know of any previous work that is concerned with a mapping from answer phrase features to question phrases for German. The rules presented in the following are to be understood as provisional, based solely on my linguistic intuitions, hand-crafted examples and intermediate system outputs for some real-life data.

4.2.1. Nominal Phrases

From a linguistic point of view, we need only two features to generate question phrases for most noun phrases (except for some special cases like predicative expressions and adverbial noun phrases, see below): The grammatical function or the grammatical case of the noun phrase and its semantic category. In practice, however, the tools that automatically annotate these two features are not perfect, and having some redundant

	Feature quadruple	Question phrase
1	(Nom, SB, Person, ∗)	wer
2	(Nom, SB, Group, ∗)	wer
3	(Nom, SB, ∗, ∗)	was
4	(Nom, PD, ∗, ∗)	was
5	(Nom, SB, ∗, Kommunikation)	wer
6	(Nom, SB, ∗, Kognition)	wer
7	(Acc, SB, Person, ∗)	wer
8	(Acc, SB, ∗, Kommunikation)	wer
9	(Acc, SB, ∗, Kognition)	wer

Table 4.2.: Question phrases for nominative noun phrases.

features from different sources might compensate for some of these imperfections. I use a feature quadruple: The grammatical case, the grammatical function and the semantic category or group of the noun phrase plus the semantic category of the verbal semantic head of the sentence. By 'verbal semantic head of the sentence' I mean the verbal lexical element whose valency determines the subject and the objects we observe in a given sentence. This usually is the dependency head of any given nominal phrase (identified by the MATE parser). If the dependency head is an auxiliary, e.g., in analytical verb forms, the system looks for an infinitive or past participle dependent of this auxiliary instead; if there is no such element, e.g., in predicative constructions, there is no verbal semantic head.

Nominative Noun Phrases

Table 4.2 lists the rules for nominative noun phrases[7]. The first four rules, are the only rules strictly necessary. For subject (SB) nominative (Nom) NPs the system generates *wer* or *was* depending on whether they refer to a person or not. Groups (e.g., companies, organizations) are a special case, since they only seem to be treated like persons in subject position (and as oblique objects), see the questions in (16), which all ask for *die Regierung*.

(16) a. Wer hat das Gesetz eingebracht?
b. ??Was hat das Gesetz eingebracht?
c. ?Wen mag er nicht?
d. Was mag er nicht?

The person/non-person distinction is not made for predicatives (PD), see (17).

[7]Question phrases are actually represented as syntactic trees. For the sake of readability and space, I only give them as strings.

	Feature quadruple	Question phrase
1	(Acc, OA, Person, ∗)	wen
2	(Acc, OA2, Person, ∗)	wen
3	(Acc, OA, ∗, ∗)	was
4	(Acc, OA2, ∗, ∗)	was
5	(Acc, MO, Menge, ∗)	wie weit
6	(Acc, MO, Durative, ∗)	wie lange
7	(Acc, MO, Iterative, ∗)	wie oft
8	(Acc, MO, Punctual, ∗)	wann

Table 4.3.: Question phrases for accusative noun phrases.

(17) Petra ist (eine) Lehrerin.

a. ?Wer ist Petra?

b. Was ist Petra?

Subjects of nominal plural subject predicatives are excluded up front to avoid questions like (18-b). Potentially good questions like *Wer ist Lehrerin?* are generated, although they may be vague (e.g., if the text mentions two teachers).

(18) Die meisten Passagiere von Flug MH370 waren Chinesen.[8]

a. Was waren die meisten Passagiere von Flug MH370?

b. *Wer waren/war Chinesen?

The remaining rules make up for some frequent annotation deficiencies: The named entity recognizer sometimes does not recognize surnames (the problem with first names was already fixed, see section 3.6). Since verbs of saying (*Kommunikation*) and thinking (*Kognition*) appear quite often, I decided to fix the semantic class errors in these cases by assuming that only humans speak and think (rules 4 and 5, Table 4.2). Occasionally, people write about talking and thinking animals, especially pets, but these people maybe also prefer to use *wer* when asking for these animals. The RFTagger sometimes confuses nominative and accusative – in case of mismatching annotations, I trust the dependency parser (rules 6–8, Table 4.2).

Accusative Noun Phrases

The rules for accusative noun phrases are listed in Table 4.3. Here, we have the same distinction between persons and non-persons as for nominatives. Accusative noun phrases sometimes have the grammatical function of temporal (19-b)–(19-d) or

[8] "Angespültes Flugzeuteil stammt von Boing 777" (Reuters, August 2, 2015).

local (19-a) adverbials (*adverbial accusatives*). If they appear as temporal adverbials, accusative noun phrases can further be distinguished by the way they modify the aktionsart of the predicate[9]. I identified three subgroups that require different question phrases: Durative (19-b), iterative (19-c) and punctual (19-d) adverbial noun phrases.

(19) a. 100 Meter, zehn Millimeter
b. zwei Stunden, einige/viele Wochen, ein paar Jahre, den ganzen Tag
c. fünf Tage die/pro Woche, jeden Tag, alle drei Stunden
d. letzten/nächsten Monat, diese Woche

Simply associating each group with a different question phrase (as rules 6–8 in Table 4.3 do) does not always lead to optimal results. The declarative sentence in (20) is an acceptable answer for questions (20-a) and (20-b), but (20-c) probably is a more fitting question under discussion. For now, I content myself with the simple model leading to slightly suboptimal results.

(20) Er wäscht sein Auto jeden Sonntag.
a. Wie oft wäscht er sein Auto?
b. Wann wäscht er sein Auto?
c. Wann wäscht er (für gewöhnlich) immer sein Auto?

Fine-grained temporal distinctions are not directly available from the automatic annotation. The grammatical function only tells us whether a noun phrase is a modifier, its semantic class whether the head of the phrase is a noun from GermaNet's semantic fields *Zeit* (19-b)–(19-d) or *Menge* (19-a)[10]. For all modifier noun phrases whose lexical head is not from the semantic field *Menge*, the system tries to identify temporal subgroups with the heuristics in Table 4.4. Each heuristic rule refers to morpho-syntactic features of the first word in the phrase and the number feature of the lexical head (both obtained from morphologically rich tags) as well as the position of the lexical head in the constituency parse (whether it is the last terminal or not). A noun phrase is a punctual adverbial if it has a singular lexical head and its first word is classified as adjective (e.g., *nächster*, *letzter*) or demonstrative pronoun (e.g., *dieses*) by the tagger (rules 7 and 8, Table 4.4). Cardinals, indefinite articles and what the STTS-tagset calls 'attributive indefinite pronouns without determiner' (e.g., *jede*, *einige*, *viele*) appear as first words in both iterative and durative noun phrases. I assume that durative nominal phrases that start with a cardinal number or an indefinite article need a right-peripheral lexical head (rules

[9]Here I assume a broad definition of aktionsart referring to the temporal structure of a (possibly complex) predicate rather than mere lexical aspect.

[10]Even this classification sometimes fails, for example, *ein paar Jahre* appeared as an accusative object in the dependency parse, *Stunde* has been misclassified as a noun from the semantic field *Kommunikation* because the word sense disambiguation component picked the synset of *Unterrichtsstunde*.

	First word	Lexical head		Subgroup
		Number	Right-Peripheral?	
1	CARD	Pl	No	Iterative
2	ART, Indef	∗	No	
3	PRO, Def|Indef, Attr	Sg	∗	
4	CARD	Pl	Yes	Durative
5	ART	Sg	Yes	
6	PRO, Indef, Attr	Pl	Yes	
7	ADJA	Sg	∗	Punctual
8	PRO, Dem, Attr	Sg	∗	

Table 4.4.: Heuristics for determining temporal NP subgroups.

4 and 5 in Table 4.4, cf. the examples in (19-b)), whereas iterative noun phrases require extra material after the head (rules 2 and 3 in Table 4.4, cf. the first example in (19-c)). There are two distinct subsets of attributive indefinite pronouns that appear exclusively in either durative or iterative noun phrases, but the tagset does not distinguish them. To identify the correct temporal subgroup in these cases, the number feature of the lexical head is used (rules 3 and 6, Table 4.4).

Dative and Genitive Phrases

The mapping from dative and genitive nominal phrases is shown in Table 4.5. For genitives and datives we again have the person/non-person distinction, but there is no impersonal question word. To avoid misleading questions, the system only generates question words for phrases referring to persons. Postnominal genitive modifiers often falsely appear as independent clause-level phrases in the constituency parse. To prevent asking for these phrases, genitives are also required to be labeled as genitive objects (OG) by the dependency parser.

	Feature quadruple	Question phrase
1	(Dat, ∗, Person, ∗)	wem
2	(Dat, ∗, Group, ∗)	wem
3	(Gen, OG, Person, ∗)	wessen
3	(Gen, OG, Group, ∗)	wessen

Table 4.5.: Question phrases for dative and genitive noun phrases.

4.2.2. Prepositional Phrases

To generate question phrases from prepositional phrases, the system has access to five features that might be relevant: The string of the prepositional phrase's head, the grammatical function of the prepositional phrase, the grammatical case and the semantic category of the nominal head of the noun phrase governed by the preposition and the semantic class of the verbal semantic head of the sentence. Since there are so many rules for prepositional phrases, I will only describe some common patterns and give one complete example to illustrate them. The complete list of rules can be found in Appendix B.2. To handle contracted prepositions like *aufs* or *zum*, both the full preposition string and a truncated version are matched against the rules; *im* is treated separately.

Table 4.6 contains all rules for *auf*, a preposition governing accusative and dative case. The choice of the correct question phrase for prepositions governing accusative and

	Feature quintuple	**Question phrase**
1	(auf, Acc, Person, ∗, ∗)	auf wen
2	(auf, Acc, ∗, MO, ∗)	wohin
3	(auf, Dat, Person, ∗, ∗)	auf wem
4	(auf, Dat, ∗, MO, ∗)	wo
5	(auf, ∗, ∗, OP, ∗)	worauf

Table 4.6.: Question phrases for prepositional phrases with *auf* as head.

dative usually depends on three features, namely grammatical case, semantic class and grammatical function. For persons, the question phrase always consists of the preposition and the question word asking for the noun phrase governed by the preposition (rules 1 and 3, Table 4.6; examples (21-a) and (21-d)). For non-persons, the distinction between prepositional objects (OP) and modifiers (MO) is important if the preposition is a potential head of a prepositional object[11]. The latter are further distinguished according to grammatical case (for *auf*, there is no further semantic distinction, the accusative question phrase always asks for a directive modifier, the dative question phrase for a local modifier; rules 2 and 3, Table 4.6; examples (21-c) and (21-f)); question phrases for the former contain the preposition and start with *wo*, independent of grammatical case (rule 5, Table 4.6; examples (21-b) and (21-e); only in colloquial German and for some prepositions in standard German (e.g., *neben*) we instead have 'preposition + *was*').

(21) a. Der Bankräuber schießt auf den Polizisten.
b. Der Schauspieler verzichtet auf seine Gage.

[11] The TIGER annotation scheme lists 14 prepositions that can be heads of prepositional objects (Albert et al., 2003, p. 57).

c. Er rennt auf die Straße.
d. Viel Druck lastet auf der Produzentin.
e. Der Film beruht auf einer wahren Begebenheit.
f. Ein Mann steht auf der Straße.

Genitive prepositional phrases each have exactly one question phrase, usually of the form 'preposition + *wessen*'. Some exceptions are *während* (*wann*), *wegen* (*weswegen*), *diesseits* and *außerhalb* (*wo*). For almost all remaining prepositional phrases (those with prepositions governing the dative, genitive and dative or accusative case) there are separate rules for persons and non-persons. For *nach* and *zu* also the object/modifier distinction is relevant. Modifiers can be either local (*wo*), directional (*wohin*) or temporal (*wann*).

The last feature of the quintuple currently is not used. At first sight, it seems like the semantic class of the verb could help to distinguish between cases like (22-a) and (22-b), if we assume that prepositional phrases headed by *zum* have directional semantics for verbs that involve movement. However, example (22-c) shows that this is not true in general.

(22) a. Ich laufe zum Strand. (Wohin ...?)
b. Das Meer gehört zum Strand. (Wozu ...?)
c. Ich laufe zum Spaß/Vergnügen. (Wozu/Warum ...?)
d. Ich laufe zum Fenster. (Wohin ...?)

A more promising approach seems to be to distinguish between prepositional objects like (22-b) and modifiers like (22-a) and (22-c). Modifiers then can be further divided according to semantic properties of their noun phrases. In (22), the question word *wohin* only appears with noun phrases denoting locations (22-a) or at least concrete objects with spatial extension (22-d).

4.2.3. Subordinate Clauses

Embedded clauses as a whole can be targets for questions. The question phrase depends on the grammatical function of the clause in the sentence. Traditional grammar distinguishes four broad functions of subordinate clauses: They may appear as subjects, objects, adverbials or attributes. To find the correct question phrase, each of these four groups must be further distinguished into a variety of subgroups. The problem is that the features used in finding question phrases for nominal phrases and prepositional phrases are not of much help here: 1) Clause markers are often ambiguous. Trying to identify the meaning of a subordinate clause based solely on the first word only works for some subordinating conjunctions introducing adverbial clauses (e.g., *weil* and

seit). 2) The grammatical function of clauses is not marked morphologically, so there is no case feature (with the exception of free relatives, see below). 3) The grammatical functions assigned to subordinate clauses (or more precisely, to the heads of subordinate clauses) by the dependency parser are not useful – they are labeled as clausal objects or modifiers according to whether the subordinating conjunction introduces a verb-second or a verb-last clause (cf. Albert et al., 2003, p. 50). 4) The constituency parses often contain unexpected structures. Non-finite subordinates usually appear as verbal phrases in the constituency parse, however, if a correlative element immediately precedes them, they are wrapped in prepositional phrases if the correlative element is a pronominal adverb (figure 4.1a) or nominal phrases if the correlative element is *dessen*. These two problems can be dealt with because the resulting structures are unique, but other cases of parsing failures are not so easily recoverable. For example, if there is a correlative *es* in the midfield and both left and right bracket are occupied, the parser assumes that *es* is a subject pronoun and we get a coordination of verbal phrases. These structures also occur in other contexts, in which they are the correct analysis, so we cannot simply design a special rule for them.

In the following, I give a brief overview of different forms, functions and meanings of subordinate clauses. This overview roughly follows the traditional classification of embedded clauses in German as outlined in the grammar of Helbig and Buscha (1987, pp. 653–695), but I add some distinctions that I deem relevant with respect to the task and omit others that are not important in this context. Referring to the overview, I will single out problematic cases and describe how the system deals with embedded clauses.

Subject Clauses

Subject clauses can occur in prototypical subject position in the prefield (23-a), or extraposed to the postfield[12]. If the subject clause is extraposed and the prefield is not occupied by another constituent, there must be a correlative element (either *es* or *das*[13]) in the prefield (23-b), otherwise, there may sometimes be an optional correlative element in the midfield (23-c) or, immediately following the clause, in the prefield (23-a).

(23) Subject clauses

[12]Subject clauses (as well as object and adverbial clauses) may also appear in a position before the prefield, see example (i) taken from Pittner and Berman (2007, p. 109), but for now, I ignore these cases.

(i) Dass sie gut vorlesen kann, das beeindruckt ihn sehr.

[13]Helbig and Buscha (1987) also list semantically underspecified nouns (or noun phrases) like *die Tatsache* as possible correlatives, but I will treat them as heads of attributive clauses to prevent the problem of having to distinguish deverbal nominalizations like *Hoffnung* from semantically underspecified nouns like *Tatsache*.

a. Ob der Kommissar den Mörder findet, (das) wird sich bald zeigen.
b. Es wird sich bald zeigen, ob der Kommissar den Mörder findet.
c. Bald wird (es) sich zeigen, ob der Kommissar den Mörder findet.
d. Dass es regnet, stört ihn nicht.
e. Es interessiert ihn, wie Kekse hergestellt werden.
f. Seinen Freund wiederzusehen, freut ihn sehr.

Subject clauses can be finite, introduced by a subordinating conjunction (*dass* or *ob* (23-a)–(23-d) or an interrogative adverb like *wie* (23-e)), or non-finite (23-f). All subordinate clauses in (23) answer the question word *was*. A special case are so-called 'free' or 'nominal' relative clauses like those in (24).

(24) Free relatives in subject position

a. Was du willst, interessiert mich nicht.
b. Wer nichts zu verbergen hat, muss nichts fürchten.
c. Wem das nicht passt, kann ja gehen. (*Wem kann gehen?)
d. Erforscht wird, wofür's Geld gibt. (*Wofür wird erforscht?)

In the first two examples, the relative pronoun is identical to the question word, however, the examples in (24-c) and (24-d), which I copied from Sternefeld (2007, pp. 391f.), show that this is not always the case. The nominative required for subjects is not always reflected in the relative pronoun of free relatives (this is also true for direct object free relatives and the accusative case, see below).

Direct Object Clauses

If we replace object clauses with pro-forms, we can see that they need to be distinguished further according to object type. If the clause acts as a direct object, the corresponding question phrase is *was*. Direct object clauses can have different forms: They can be verb-last clauses introduced by subordinating conjunctions (25-a)–(25-c), unintroduced verb-second clauses (25-d), non-finite subordinates (25-e) or free relatives (26).

(25) Direct object clauses

a. Peter glaubt, dass Maria ihn betrügt.
b. Peter fragt sich, ob Maria ihn betrügt.
c. Er beobachtet (?es), wie eine Möve ein Kaninchen verschlingt.
d. Er sagt, er habe sie nicht gesehen.
e. Er hat (es) bedauert, nicht da gewesen zu sein.

Sometimes, the extraposed clause in the postfield is accompanied by a correlative element in the midfield, for example, in (25-c) and (25-e). The relative pronoun of free relative clauses that appear as direct objects can be identical to the sought-after question word (26-a), but this is only due to chance as examples (26-b)–(26-d) show (Sternefeld, 2007, p. 391). If the matrix clause verb requires the accusative case, this apparently need not be reflected in the free relative.

(26) Direct object free relatives

a. Er weiß, was er tut. (Was weiß er?)
b. Ich suche aus, wem ich mich unterwerfe. (*Wem suche ich aus?)
c. Er zerstört, wovon er abhängig ist. (*Wovon zerstört er?)
d. Jeder muss tun, wofür er bestimmt ist. (*Wofür muss jeder tun?)

Genitive and Dative Object Clauses

The next type of clausal objects can be substituted with a genitive pro-form (*wessen*). These clauses can be non-finite (27-a) or finite verb-last clauses introduced by a subordinating conjunction (27-b).

(27) Genitive and dative object clauses

a. Er hat mich (dessen) beschuldigt, ihn geschlagen zu haben.
b. Er hat sich (dessen) vergewissert, dass die Frau tot war.
c. Bodo entledigt sich, wessen er nicht mehr bedarf.
d. Ich folge, wem immer ich vertraue.

Again, in some cases a correlative element may appear in the midfield, although the sentences with *dessen* to me feel a bit dated and I would expect them to appear less frequent today. In contrast to the previous cases, if a verb governs the genitive, free relatives need to be congruent (27-c); the same is true for verbs governing the dative (27-d) and in this case, free relatives are the only clausal option that I could think of (both examples are taken from Sternefeld (2007, p. 391)).

Prepositional Object Clauses

The last group of object clauses function like prepositional phrases, and actually the Stanford parser also labels non-finite clauses like (28-a) as PPs with the pronominal adverb as the head, see figure 4.1a. They can occur in the postfield, but never in the midfield or prefield. Prepositional object clauses can be finite (28-c)–(28-e), introduced by a subordinating conjunction, or non-finite (28-a), (28-b).

(28) Prepositional object clauses

a. Peter träumt (davon), ein Auto zu besitzen.
b. Peter droht (damit), sich in die Luft zu sprengen.
c. Peter prahlt (damit), dass er viel Geld besitzt.
d. Er hat sich (darüber) geärgert, dass er durch die Prüfung gefallen ist.
e. Er verlässt sich darauf, dass wir pünktlich sind.

For both types, there may be a pronominal adverb as correlative element in the midfield. Depending on the verb, this correlative element is optional (28-a)–(28-d) or obligatory, see example (28-e), copied from Helbig and Buscha (1987, p. 671)[14]. The versions without pronominal adverb can sometimes be elicited with *was*-questions, but in many cases questions with interrogative pronominal adverb are preferred:

(29) a. Wovon/Was träumt Peter?
b. Womit/??Was droht Peter?
c. Womit/?Was prahlt er?
d. Worüber/?Was hat er sich geärgert.
e. Worauf/*Was verlässt er sich?

The *was*-question in (29-b) is especially suspect, because it introduces an ambiguity between Peter being the subject or an object, *was*-questions for prepositional clauses with obligatory correlative elements are ungrammatical. Prepositional object clauses without pronominal adverb look very much like subject and object clauses and can, in fact, not be distinguished from them by the system. Hence, if they do not allow for *was*-questions, there is a problem. If we can identify a pronominal adverb which refers to a prepositional object clause in the postfield, finding the question phrase is easy. Pronominal adverbs that refer to an extraposed prepositional clause always start with *da* (e.g., *dafür*, *damit*, *daran*). Replacing *da* with *wo* yields the correct question word.

Adverbial Clauses

Adverbial clauses modify the main verb or the whole proposition of their matrix clause. According to the way they do this, adverbial clauses can be classified into a number of different semantic types. They usually have the form of finite or infinite clauses introduced by a subordinating conjunction, sometimes with optional or obligatory correlative element (depending on the conjunction), and may appear in the prefield, the midfield or the postfield. Table 4.7 lists some types together with a selection of possible subordinating conjunctions (and correlative elements, which are set off by a comma and

[14]The prepositional clause and its correlate may occupy the prefield together, which is not possible with the correlate *es* of subject or direct object clauses and suggests an attributive analysis (Pittner & Berman, 2007, p. 111)

Type	**Subordinating conjunctions**	**Question word**
temporal	bevor; bis; (als), *da*; *nachdem*; seit; *solange*; sobald; sowie; *während*; *(dann), wenn*	(seit/bis) wann
local		
place	wo	wo
origin	*woher*	woher
direction	*woher*; wohin	wohin
modal		
instrumental	indem	wie
	damit/dadurch, dass	womit/wodurch
restrictive	(in)*sofern*; (in)soweit	inwieweit/inwiefern
comparative	(so), wie; als	?
causal		
causal (s.s.)	*nachdem*; (daher/deshalb/deswegen), weil/*da*	warum
conditional	*solange*; *(dann), wenn*/falls/*sofern*	(?) wann
concessive	obgleich; obschon; obwohl; *wenn* ... auch	?
consecutive	sodass; ohne dass; genug, um ... zu	?
final	damit; um ... zu; (auf) dass	wozu
substitutive	(an)statt dass	?
adversative	indes(sen); *während*; wohingegen	?

Table 4.7.: Types of adverbial clauses.

enclosed in parentheses if they are optional) and a simple substitutive question phrase, if there is one. The table is by no means exhaustive and the selection of subordinating conjunctions and correlative elements is more or less random, but still it illustrates two of the main problems in generating question phrases for adverbial sentences. Some subordinating conjunctions may introduce adverbial clauses of different types (forms that appear twice in Table 4.7 are written in italics) and require different question phrases. The example sentences in (30) show the difference between a temporal and an adversative clause introduced by *während*.

(30) a. Während er auf Arbeit ist, räumen ihm Einbrecher die Wohnung aus.
b. Während Maria die Beatles mag, hört Peter lieber die Rolling Stones.
c. Während Epikur seine Jünger in einem Garten um sich versammelte, unterhielt sich Sokrates mit den Menschen auf dem Marktplatz von Athen.

Distinguishing between different types of adverbial sentences with the same clause marker can be hard, it may require information about tense and aspect, the compositional semantics of a clause, its context or even world knowledge (to know that the adverbial

clause in (30-c) cannot be temporal, one needs to know that Socrates was dead before Epicurus founded his school).

The second problem is indicated by the question marks in Table 4.7. For some types of adverbial clauses, there are no question phrases that can simply be substituted for the clause in order to form a good question. A possible question phrase for the concessive clause in (31-a) might be *wessen ungeachtet* or *welchem Umstand zum Trotz*, but the resulting questions sound very contrived (32-a). The question in (32-b) for the adversative clause in (31-b) is completely vague.

(31) a. Peter hat Angst, obwohl es dafür keinen Grund gibt.
b. Maria studiert, wohingegen er eine Ausbildung macht.

(32) a. Wessen Ungeachtet/Welchem Umstand zum Trotz hatte Peter Angst?
b. Was steht im Gegensatz dazu, dass Maria studiert?

The cases that are easy to handle are those were there is a one-to-one relation between subordinating conjunction and question phrase:

(33) a. Wenn sie in der Stadt ist, besucht sie ihn. (Wann . . . ?)
b. Seit sie aufs Land gezogen ist, geht es ihr besser. (Seit wann . . . ?)
c. Ich warte dort (deshalb) auf dich, weil ich dich mag. (Warum . . . ?)

Attributive Clauses

Attribute clauses are modifiers of nouns or noun phrases. To this category belong relative clauses (34) and some clauses that refer to semantically underspecified nouns (36), like *Tatsache* and *Fakt*, or to nouns that are products of deverbal nominalizations (37), like *Frage* and *Hoffnung*.

Relative clauses can be restrictive (34-a) or non-restrictive (34-b).

(34) a. Er liest das Buch, das er gestern gekauft hat.
b. Die Tübinger Universität, die bereits 1477 gegründet wurde, hat heute fast 30.000 Studenten.

Since a restrictive relative selects an element or a subset of the set that is the extension of the noun or noun phrase it modifies, we can ask questions like (35-a). A similar question is not possible for non-restrictive relative clauses: (35-b) falsely implicates that there was more than one university in Tübingen[15].

[15]In Gricean terms, we could speak of a violation of the cooperative principle: If there was only one university, the questioner would prompt his conversation partner to make a contribution that is not informative (violating the maxime of quantity), hence the addressee of question (35-b) has to assume that there is more than one university.

(35) a. Welches Buch liest er?
b. *Welche Tübinger Universität hat heute fast 30.000 Studenten?

In English, orthographic rules distinguish restrictive and non-restrictive relative clauses (only the latter are set off by commas). Telling them apart in German, however, is a non-trivial problem, which the system currently does not solve. To avoid questions like (35-b), relative clauses are ignored completely.

Questions for attributive clauses referring to underspecified nouns can have the same structure as those for restrictive relative clauses (36-a). Probably even better, however, are questions like (36-b), which can be explained with an analysis where *Tatsache* is a correlative element of the clause (like *es* or *das*) that is purely functional and has no semantic value at all.

(36) Den Autor freute die Tatsache, dass nach der Lesung Kritik geübt wurde.
a. Welche Tatsache freute den Autor?
b. Was freute den Autor?

Attributive clauses referring to deverbal nominalizations also allow for two different types of questions: A question parallel to those in (35-a) and (36-a) that is based on the morpho-syntactic properties of the lexical head, which identify it as a noun (37-a), and a question that is based on the semantic properties of the head, which allow for a verbal alternative (37-b).

(37) Er hat die Hoffnung, dass sich sein Leben ändern wird.
a. Welche Hoffnung hat er?
b. Was hofft er?

Problem Summary

It is not always easy to distinguish non-relative subject and direct object clauses. They can have the same form, both appear in the prefield and the postfield and the correlative elements *es* and *das*, which only appear in the prefield of subject clauses and hence could be used to identify them, also appear as pro-forms in other contexts. This is not really problematic, because the question word for both is *was*. Relative clauses in subject and direct object position need not bear nominative or accusative case, thus a relative clause introduced by a *w*-word bearing genitive or dative case can have the function of a subject, accusative object, genitive object or dative object.

Often prepositional clauses come without a correlative element in the midfield (38-a). In these cases, they are hard to tell apart from subject and direct object clauses (38-b).

(38) a. Er hat sich geärgert, dass er durch die Prüfung gefallen ist.
b. Er hat sich eingeredet, dass er durch die Prüfung gefallen ist.

This is problematic, since not all prepositional clauses allow *was*-questions and even if they are not ungrammatical, they are usually suboptimal (39).

(39) ?Was hat er sich geärgert?

Question phrases for prepositional clauses can be generated if the system can identify a pronominal adverb in the midfield that refers to the prepositional clause. However, proniminal adverbs in the midfield can also refer to the preceding context (40-a).

(40) a. Er hat damit gezeigt, dass er ihm Schaden zufügen kann.
b. Er hat sich damit abgefunden, dass er ins Gefängnis muss.
c. Er hat sie damit beruhigt, dass er ihr ein Schlaflied vorsang.

Without information about the context or the valency of *zeigen*, the direct object clause in (40-a) will be mistaken for a prepositional clause (40-b) or an instrumental adverbial clause (40-c). The result will be the question in (41).

(41) *Womit hat er gezeigt?

Adverbial clauses with ambiguous clause markers can have a range of semantic types, which allow for different types of questions. Disambiguating the semantic type often is not feasible, as shown above.

Attributive clauses pose two problems: Restrictive relatives cannot be distinguished from non-restrictive relatives (which leads to questions like (35-b)) and attributive clauses introduced by *dass* or *ob* (42-a) can look very much like object clauses (42-b).

(42) a. Peter äußert die Hoffnung, dass es Überlebende geben könnte.
b. Peter lehrt die Kinder, dass zwei plus zwei vier ist.

A special challenge are non-adjacent attributive clauses, that is, attributive clauses extraposed to the postfield in a sentence where the head is in the prefield, or the head is in the midfield, and the right bracket is not empty:

(43) a. Die Frage ist interessant, die Sie gestellt haben. (Pittner & Berman, 2007, p. 114)
b. Peter hat die Hoffnung geäußert, dass es Überlebende geben könnte.

Suboptimal Solution

Looking at the little typology of embedded clauses and the summary of problematic cases above, we can see that, given the linguistic information currently available, the only embedded clauses for which the system has a chance of reliably generating good questions are unambiguously marked adverbial clauses. This only requires a simple lookup table that returns a question phrase for each unambiguous clause marker. To cover some more interesting cases, I decided to overgenerate, that is, to generate multiple questions for each ambiguous clause of which at least one is correct.

In a first step, the system tries to identify potential correlative elements. The sentences in (44) tell us different things about Peter, but the important linguistic property they share is that they all have a pronominal adverb in the midfield and a subordinate clause in the postfield.

(44) a. Peter träumt davon, ein Auto zu besitzen.
b. Peter hat davon geträumt, ein Auto zu besitzen.
c. Peter prahlt damit, dass er viel Geld besitzt.
d. Peter hat damit geprahlt, dass er viel Geld besitzt.
e. Peter hat darüber reden wollen, wie es weiter geht.
f. Peter will Maria damit zeigen, dass er sie liebt.

In the first five sentences, the subordinate clause is a prepositional object clause, which the pronominal adverb in the midfield refers to. The first two of them are non-finite, the next three are finite. In examples (44-a) and (44-c), the right bracket of the matrix clause is empty, which puts the pronominal adverb in an adjacent position to the embedded clause. Changing the tense of the matrix clause to present perfect results in a filled right bracket between the pronominal adverb and the embedded clause in examples (44-b) and (44-d). The right bracket need not always be filled only by a past participle, but can be more complex, as example (44-e) shows. Sentence (44-f) looks very similar to (44-d), but the pronominal adverb does not refer to the embedded clause, which is not a prepositional object clause, but an accusative object clause. Figure 4.1 contains the Stanford parses[16] for the first four prepositional object clause sentences. All four parses show very different syntactic structures.

In the parse in Subfigure 4.1a, the pronominal adverb and the embedded non-finite clause form a prepositional phrase that is labeled as prepositional object by the dependency parser. Since prepositional phrases are treated as potential answer phrases, all constituents dominated by them are subject to movement constraints. Instead of dealing

[16]The terminal categories are basic tags obtained from the RFTagger – I deleted all other features attached to the nodes to save space, except for grammatical functions of immediate descendants of S nodes.

```
(ROOT
  (S (N-SB Peter) (VFIN träumt)
    (PP-OP (PROADV davon) (SYM ,)
      (VP
        (NP (ART ein) (N Auto))
        (VZ (PART zu) (VINF besitzen))))
    (SYM .)))
```

(a) Non-finite subordinate, empty right bracket.

```
(ROOT
  (S (N-SB Peter) (VFIN-hat)
    (CVP-OC
      (VP (PROADV davon) (VPP geträumt))
      (SYM ,)
      (VP
        (NP (ART ein) (N Auto))
        (VZ (PART zu) (VINF besitzen))))
    (SYM .)))
```

(b) Non-finite subordinate, filled right bracket.

```
(ROOT
  (S (N-SB Peter) (VFIN prahlt)
     (PROADV-MO damit) (SYM ,)
    (S-OA (CONJ-CP dass) (PRO-SB er)
      (NP-OA (PRO viel) (N Geld))
      (VFIN-RE besitzt))
    (SYM-Pun-Sent .)))
```

(c) Finite subordinate, empty right bracket.

```
ROOT
  (S (N-SB Peter) (VFIN hat)
    (VP-OC
      (PP (PROADV damit))
      (VPP geprahlt) (SYM ,)
      (S (CONJ-CP dass) (PRO-SB er)
        (NP-OA (PRO viel) (N Geld))
        (VFIN-MO besitzt)))
    (SYM-Pun-Sent .)))
```

(d) Finite subordinate, filled right bracket.

Figure 4.1.: Stanford parses for the first four sentences in (44).

with the verbal phrase that is the root of the non-finite subordinate, we have to address the whole prepositional phrase: Whenever the first terminal of a prepositional phrase is a pronominal adverb that starts with *da*, the system obtains the question phrase from the pronominal adverb by replacing *da* with *wo*.

The parse of the second sentence (Subfigure (44-b)) is problematic because it is not unique to this kind of sentences. Coordinated verbal phrases might also contain pronominal adverbs in other cases and asking for the second verbal phrase in these coordinations is certainly going to result in ill-formed output. This is why currently no questions are generated for non-finite subordinates after filled right brackets.

Parses 4.1c and 4.1d have some structural parallels. The root of the subordinate clause is an S node and it is not dominated by any potential answer phrase. Here, the system looks for a comma[17] and a correlative element in the four terminals preceding the embedded clause. If it finds a pronominal adverb, it is used to generate a question phrase as described above and the search terminates. For (44-d) and (44-f) this lead to the questions in (45-a) and (46-a).

(45) Peter hat damit geprahlt, dass er viel Geld verdient.

a. Womit hat Peter geprahlt?

[17]The comma is marked and will be filtered out upon generating a new question.

Subordinating conjunction	**Question phrase**
bis	bis wann
dass, ob	was
falls, nachdem, sobald, wenn	wann
seit, seitdem	seit wann
sofern	unter welcher Bedingung
solange	wie lange
sooft	wie oft
weil	warum

Table 4.8.: Mapping from subordinating conjunctions to question phrases.

b. *Was hat Peter damit geprahlt?

(46) Peter will Maria damit zeigen, dass er sie liebt.

a. *Womit will Peter Maria zeigen?

b. Was will Peter Maria damit zeigen?

As soon as a terminal is encountered that is a finite verb (the end of the midfield) or neither a verb nor a comma (an intervening non-verbal constituent), the search terminates as well. If the subordinating conjunction of an embedded clause is either *dass* or *ob* and the system finds a comma preceded by a noun of the semantic group *Abstract_Entity*, it generates a question for an attributive clause. For (42-a) and (42-b) this lead to the questions in (47-a) and (48-a).

(47) Peter äußert die Hoffnung, dass es Überlebende geben könnte.

a. Welche Hoffnung äußert er?

b. *Was äußert Peter die Hoffnung.

(48) Peter lehrt die Kinder, dass zwei plus zwei vier ist.

a. *Welche Kinder lehrt Peter?

b. Was lehrt Peter die Kinder?

Independent of what questions have been generated previously, at the end, the subordinating conjunction is looked up in Table 4.8 and another question is generated if there is a matching question phrase. This leads to the second questions in (45)–(48).

It might be possible to avoid generating some bad questions even with the limited information at hand, e.g., we could define an exhaustive list of deverbal nominalizations and semantically underspecified nouns that allow for attributive *dass*-clauses to avoid generating *was*-questions in these cases, but to fully resolve all ambiguities, at least information about the argument frame of the matrix verb would be necessary.

4.2.4. Other Units

Apart from noun phrases, prepositional phrases and embedded clauses, there are other elements in a sentence that could be asked for. This section mentions some answer units the system does not cover.

Possessives

Possessives should be easy targets for questions: One simply has to replace them with *wessen* before moving the whole noun phrase or prepositional phrase they belong to into the prefield, see the examples in (49)[18].

(49) a. Der Journalist befragt Merkels Sprecher.
b. Wessen Sprecher befragt der Journalist?
c. Der Journalist richtet eine Frage an den Sprecher der Bundesregierung.
d. An wessen Sprecher richtet der Journalist eine Frage?

There are, however, several problems: Genitive nouns are only recognized as such by the RFTagger, if the genitive marker *-s* is separated from the word by an apostrophe, but current orthographic rules advise against such an apostrophe. If nouns consist of multiple parts, only the last part is marked and recognized as a genitive (e.g., *Peter Pan's*). Prenominal genitives and the nominal head are put under the category MPN (multi-word proper noun) by the constituency parser (e.g., *[$_{MPN}$ Peter Pan's Flucht]*), which makes it hard to identify them, especially when they are not recognized as genitives. Postnominal genitives are handled inconsistently by the constituency parser, sometimes they appear as a noun phrase under a complex noun phrase (as expected), sometimes they appear as separate clause-level constituents.

To circumvent these problems, we could ask only for pronominal possessives, but these questions often seem odd because the answer is obvious. I tried to prevent these odd questions by only considering possessive pronouns without coreferring constituents in the same sentence, but as soon as the coreference resolution fails, inacceptable question are generated, for example, an intermediate version of the system generated the question in (50-b) for the sentence in (50-a)[19].

(50) a. Im Juli verbuchten die Finanzämter nach Angaben des Bundesfinanzministeriums vom Donnerstag mit 49,3 Milliarden Euro 8,6 Prozent mehr *in ihren Kassen* als im Vorjahresmonat.
b. ?In wessen Kassen verbuchten die Finanzämter im Juli nach Angaben des Bundesfinanzministeriums vom Donnerstag mit 49,3 Milliarden Euro 8,6

[18] Here I ignore possessive prepositional phrases like in *das Auto von meiner Schwester*.
[19] "Aufschwung beflügelt Steuereinnahmen im Juli" (Reuters, August 20, 2015).

Prozent mehr als im Vorjahresmonat?

A question like (50-b) misleadingly implicates that the money could have been registered somewhere other than in the tax coffers.

Other Attributes

Apart from clauses and different forms of possessives, attributes can also come as adjectives (51-a), adverbs (51-b), participles (used like adjectives, (51-c)), prepositional phrases without possessive semantics (51-d) and infinitives (51-e); all examples are copied from Helbig and Buscha (1987, p. 597).

(51) a. der billige Stoff, das rechte Gebäude
b. das Buch hier, das Wetter gestern
c. der schreibende Arbeiter, die abgeschlossene Arbeit
d. der Glückwunsch zum Geburtstag, der Besuch am Sonntag
e. die Fähigkeit zu abstrahieren

Any attribute may narrow down the extension of the noun or noun phrase it modifies. If the (linguistic or extra-linguistic) context provides a set of alternatives, the attribute can be used to choose one of them, which allows for questions like (52-b).

(52) a. Er hat das grüne Auto zerkratzt (nicht das blaue).
b. Welches Auto hat er zerkratzt?

Adjectives and Adverbs as Predicatives

The system asks for predicatives if they are noun phrases or prepositional phrases, but not if they are adjective phrases (53-a) or adverbs (53-b).

(53) a. Ein Formel-1-Wagen ist sehr schnell.
b. Peter ist draußen.

This is due to the fact that adjective phrases and adverbs generally are not potential answer phrases for the system. We could add predicative adjectives and adverbs as potential answer phrases and use a template like (54) to ask for them.

(54) Was wird im Text über X gesagt?[20]

In the case of adverbs, we could also ask a more specific question based on their semantic subclass (55). To do this, we would need a means of determining their semantic subclass.

[20]Where X is the subject of a subject predicative or the object of an object predicative.

For adjectives GermaNet's semantic fields might be a useful starting point, for adverbs one probably would have to compile an exhaustive list for each subclass.

(55) Wo ist Peter?

We would also have to be careful not to ask for deictic adverbs like *hier*, *dort*, *gestern* or *morgen* if their point of reference is unclear.

Adjectives and Adverbs in Adverbial Function

Questions for adjectives and adverbs in adverbial function could be generated similar to questions for adverbial sentences (based on their semantic subclass). What was said about semantic subclasses and deixis for attributive adverbs also applies to adverbs in adverbial function, of course.

4.3. Undoing Topicalization

Before the question phrase can be inserted into the tree, any constituent occupying the position before the finite verb (i.e. the prefield) needs to be moved behind the finite verb. In generative terms (and taking the metaphor of syntactic movement literally), we could say that we need to undo topicalization by moving the prefield constituent back into its previous position.[21] To do this without producing ungrammatical structures, the system needs a component that models word order.

There is extensive literature on German word order, some of it is focused on the theoretical analysis of word order phenomena within a certain framework, e.g., Büring (1994) proposes a phrase structural analysis of certain midfield phenomena, Müller (1999) an analysis within optimality-theoretic syntax; other (often older) work is mainly descriptive, e.g., Lenerz (1977)[22], Hoberg (1977), Höhle (1982), Lötscher (1984), Reis (1987) and Hofmann (1994)[23], to name but a few. For my purposes, the descriptive literature was most helpful.

[21] If the prefield is empty already, we can skip this step. The prefield can be empty, if it was occupied by the answer phrase, which is simply deleted upon the creation of a new question.

[22] This is the classic on the topic; it is discussed in each of the works on German word order mentioned here, except for Hoberg (1977), which appeared in the same year.

[23] This monograph is especially relevant, since Hofmann's system of rules for the linear order of pronouns and nominal phrases in the German midfield is supposed to enhance the speech recognition component of the dialog system SPICOS, which means that she also has to consider computational linguistic constraints.

4.3.1. Unmarked Word Order

German allows for a lot of word order[24] variation, especially in the midfield, but many grammatical word orders require contexts with certain information-structural properties. These word orders usually restrict the set of elements that can be focused in a sentence. Focus, according to Krifka (2008), indicates that there is a set of alternatives of which the focused element selects one. In some cases, these alternatives can be accommodated without any special context. Mostly, however, focus selects an alternative provided by a *question under discussion* raised (explicitly or implicitly) in the context. If a focus in a sentence ignores the current question under discussion, the sentence is going to be unacceptable in that context. Thus, by restricting focus options, a word order also restricts its potential contexts.

Initially, one might wonder how focus is relevant for modeling the word order of questions generated by the system. In the focus literature, questions are usually only considered under the aspect of introducing alternatives, rather than selecting among them. And since the questions are supposed to appear after the text, we have no immediate context providing sets of alternatives. In the following, I try to give an example that shows how asking questions (after a text) involves focusing certain elements. Imagine a text about early missionaries who brought Christianity to different countries. It might mention Pantaenus in India, Saint Patrick in Ireland, Augustine of Canterbury in England and Alopen in China. Based on this text, we could ask either of the two questions in (56).

(56) a. Wer brachte das Christentum den Iren?
b. Wer brachte den Iren das Christentum?

In both questions there is a focus on *den Iren*, in the sense that this expression selects among Indian, Irish, British and Chinese people. If we imagine a slightly different text that does not only mention Christian but also Islamic missionaries, only the second question is acceptable. The reason for this is that adding a set of alternative religions leads to a complex focus in questions that ask for a missionary that brought a certain religion to a certain people. The word order of question (56-a), however, does not allow a complex focus that involves both direct object and indirect object, which is why it is not applicable in this context. Based on a text about a fictional world, where different missionaries brought different religions to the same people, we could also ask a question with a focus only on *das Christentum*. This focus, again, is possible in question (56-b), but not in question (56-a).

Since the system cannot model questions under discussion or contextually provided

[24] With the term *word order* I refer to the order of constituents associated with different sets of features, see below.

sets of alternatives, it must choose a word order that does not restrict possible contexts. Lenerz (1977, p. 27) calls the order AB of two constituents A and B *unmarked*, if its occurrence is not subject to certain testable conditions that apply to the *marked* order BA. If we think of these conditions as different contexts, the unmarked order is exactly what we need to find to generate questions like (56-b) instead of questions like (56-a). A more refined definition of (roughly) the same notion can be found in Höhle (1982), for which I give a translation in (57)[25].

(57) **Höhle's (1982, p. 131) definition of marked and unmarked word order**
If two constituent types T1 and T2 can appear in the order T1 $<$ T2 under the structural conditions C1 and in the order T2 $<$ T1 under the structural conditions C2, where C2 $\subset$ C1, then T1 $<$ T2 is the *structurally normal order* and T2 $<$ T1 is the *structurally marked order*.

Höhle's definition clarifies some vague points in Lenerz' definition. According to the definition in (57), a word order is also unmarked or normal if it has no alternative (if C2 = $\emptyset$), whereas Lenerz' definition seems to imply that AB is only unmarked if there is a word order BA that is grammatical under certain conditions. Höhle's definition also explicitly excludes the case where T1 $<$ T2 and T2 $<$ T1 are distributed complementary, that is, when C1 $\cap$ C2 = $\emptyset$. Höhle (1982) further notes that what Lenerz calls 'constituents' are actually sets of features associated with constituents which are used to refer to constituent types. These sets may, for example, include the grammatical category (e.g, nominal or pronominal), the grammatical function, the thematic role or the definiteness of a constituent. Höhle criticizes that the selection of these features seems arbitrary, which, on the one hand, made it easier to formulate rules that describe the characteristic order of two constituents, but, on the other hand, would also raise the question about the relevance of such a notion of unmarked word order, since the distinction between features that describe the constituents (T1 and T2) and features which the conditions (C1 and C2) refer to would be blurred. In the context of a question generation system, however, I find that the distinction between constituent features and testable conditions on word orders comes quite naturally: Everything that can be determined within the sentence belongs to the constituent (e.g., the afore mentioned features 'grammatical category', 'grammatical function', 'thematic role' and 'definiteness', but not the accent, see below), whereas the conditions C1 and C2 refer to the context. The last difference between Lenerz' and Höhle's definition is the difference between 'conditions' and 'structural conditions'. Under 'structural conditions' Höhle understands those conditions that refer to syntactic, morphological, logical and phonological but not pragmatic features. He

[25] I changed the variable names and write T1 $<$ T2 instead of T1 $>$ T2 to be consistent with the rules in section 4.3.4.

excludes pragmatic features (like focus) because, according to him, Lenerz did not care about them and the only purpose of his question test[26] was to facilitate acceptability judgements of sentences with different accent patterns, not different foci or rhemes. However, when introducing his question test, Lenerz (1977, p. 14) explicitly states that it should allow to determine what is theme and rheme in a sentence (by turning the sentence into an utterance with a distinctive accent pattern). What causes Höhle's observation that pragmatic factors are ignored, is Lenerz' theme-rheme distinction, which does not correspond to any modern dimension of information structure. Lenerz (1977, pp. 11–15) first defines theme and rheme similar to topic and comment, but the question test he introduces right after the first definition and uses throughout his investigation of word orders suggests that theme and rheme are actually closer to focus and background. The main difference between Lenerz' rheme and focus as it is used by Höhle or Krifka (2008) is that there is a one-to-one correspondence between primary accents and rhemes but not foci. A simple rheme is a sentence constituent marked by a primary accent; rhemes can also be complex, but then each individual sentence constituent needs to carry a primary accent. Lenerz excludes the latter, which he calls cases with 'contrastive' or 'emphatic' accent, from his word order investigations because he thinks that they allow for special word orders that are ruled out for sentences with 'normal' intonation. Thus, eliciting all possible primary accents that a sentence can receive and determining all its possible rhemes for Lenerz actually is the same. Focus, of course, cannot simply be identified with the sentence constituent(s) marked with a primary accent. According to Lenerz, the sentence in (58), with the primary accent on the first syllable of *Christentum*, can only have one rheme, namely *das Christentum* (58-a); but if we look for possible foci, we find the five different possibilities in (58-a)–(58-e).

(58) St. Patrick brachte den Iren das CHRIStentum.

a. Was hat St. Patrick den Iren gebracht? – das Christentum
b. Was hat St. Patrick für die Iren getan? – brachte + das Christentum
c. Was hat St. Patrick getan? – brachte + den Iren + das Christentum
d. Was war mit den Iren? – St. Patrick + brachte + das Christentum
e. Was ist passiert? – St. Patrick + brachte + den Iren + das Christentum

Höhle (1982, p. 134) further argues that using conditions on possible foci to determine the unmarked word order in Lenerz' way would lead to false predictions for the two example sentences in (59), where the first sentence allows for more foci than the second one, although the second one has the unmarked word order according to Lenerz.

[26] The question test contextually binds parts of the answer sentence (Lenerz, 1977, p. 12), e.g., the first question in (58) binds the direct object of the answer.

(59) a. Karl hat das Buch dem MANN gegeben.
b. Karl hat dem MANN das Buch gegeben.

I would argue that these false predictions are due to the fact that Höhle here uses accent as a characteristic feature of a constituent, which means that the sentence in (60) has a different word order than the one in (59-b) because the constituent *dem Mann* (without accent) is not the same as the constituent *dem MANN* (with the primary accent of the sentence).[27]

(60) Karl hat dem Mann das BUCH gegeben.

This is not what Lenerz understands under a word order. For him, (59-b) has the same word order as (60). I would argue that accent in this context is secondary to information structure and not a characteristic feature of a constituent type. Thus, a sentence with a certain word order can be seen as a set of sentences with the same word order but different accent patterns. To assess the focus potential of a sentence with a certain word order, we need to form the union of the sets of possible foci for each grammatical accent pattern of the sentence. To do this, we can use Lenerz' question test, but we need to ask for any potential focus, not just for one constituent per primary accent. The sentence with the word order that can serve as an answer for the biggest number of questions is the sentence with the unmarked word order.

Since Lenerz only considers one focus (or rheme) per primary accent, his examples cannot provide enough evidence to claim that a certain word order is unmarked with respect to the focus condition, but, nevertheless, his transitive rules intuitively and in practice (when used to move constituents out of the prefield) seem to make sense. To explain the validity of his results, I propose the hypothesis in (61).

(61) **Hypothesis about the relation between primary accents and foci**
If the set of constituents that can receive a primary accent in a sentence S1 is a superset of the set of constituents that can receive a primary accent in a sentence S2 that only differs from S1 in its word order, then the set of possible foci of S1 is also a superset of the set of possible foci of S2.

If my hypothesis is true, it suffices to find a word order whose accentable constituents form a superset of all alternative word orders' accentable constituents to find a word order that is unmarked with respect to focus. This is exactly what Lenerz does.

A problem that I ignored so far arises when we abstract away from concrete sentences

[27] This is exactly the blurring of constituent features and conditions that Höhle criticized above: The reason for the fact that (59-a) can occur in more contexts than (59-b) is that (59-a) exhibits what Höhle (1982, p. 103) calls a *stylistically normal accent*, which allows for a maximum of different foci, whereas the sentence in (59-b) is *contextually marked* with respect to its accent pattern.

to talk about word orders as structural configurations. This is necessary if we want to use the unmarked word order to generate questions for any answer sentence. Pragmatic properties like information structure can only be assessed for concrete sentences, or if we were to be really accurate, only for the use of concrete sentences, i.e., utterances. The assumption that it is due to structural relations between constituent types that a sentence S1 with a word order O1 allows for more foci than a sentence S2 with a different word order O2 is just another hypothesis. But certainly, we would like to assume that if we had enough features to model the constituent types such that, apart from lexical semantics, the only difference between S1 and S2 were the constituent orders O1 and O2, the reason for any unexplained context-sensitive differences in acceptability had to be differences between O1 and O2.

4.3.2. Features of Constituent Types

Potential characteristic features include syntactic category, grammatical function, definiteness and thematic role. Most of Lenerz' (1977) rules are based on the syntactic category and the grammatical function of constituents. Both features interact, for example, among pronouns direct objects precede indirect objects, whereas among full noun phrases the opposite is true. From a purely linguistic point of view, one might argue that the linear order of grammatical functions should be replaced by a linear order motivated by a hierarchy of thematic roles, cf. section 4.3.4. However, thematic roles are not really well-defined and unavailable to the system, so I will stick to grammatical functions as closest approximation. Definiteness comes in two flavors: As a morpho-syntactic feature that, among other things, determines the declension paradigm of adjectives in German and as a semantic feature also called identifiability (cf. N. M. Klein, Gegg-Harrison, Sussman, Carlson, and Tanenhaus (2009) for an investigation of English definite noun phrases that do not require the referent to be uniquely identifiable). The latter is close to the information-structural notion of givenness and cannot easily be annotated automatically[28], the former can be extracted from morphologically rich tags (noun phrases with definite articles or proper nouns are definite) and may or may not be somewhat indicative of a certain information status, depending on the notion of givenness that is chosen (for example, only definite DPs can be referentially given and only indefinite DPs can be referentially new according to Baumann and Riester (2012), whereas indefinites and definites (even proper names) can be referentially new in Prince's (1981) taxonomy of given-new information). The context-dependent notion of definiteness should not be a characteristic feature of constituent types, but rather a condition in the sense described in the previous section. Whether the morpho-syntactic notion is needed to describe

[28]Since questions are generated from declarative sentences in the text, no referent should be completely new, but the exact type of givenness is difficult to assess.

the unmarked word order is disputed in the literature: Lenerz (1977, pp. 50–55) argues that rules referring to morpho-syntactic definiteness only apply to marked word orders, whereas Reis (1987, pp. 160–163) tries to show that these rules also affect the unmarked order, e.g., of indirect and direct objects if both are unaccented (62) or accented (63).

(62) a. ?Was, du hast einem Jungen das Buch geSTOHlen?
b. ??Was, du hast ein Buch dem Jungen geSTOHlen?
c. Was, du hast das Buch einem Jungen geSTOHlen?

(63) a. ?Karl hat einem JUNGen das BUCH geschenkt (und einem MÄDchen das RAD).
b. ??Karl hat ein BUCH dem JUNGen geschenkt (und ein RAD dem MÄDchen).
c. Karl hat das BUCH einem JUNGen geschenkt (und das RAD einem MÄDchen).

(62-a) and (63-a) exhibit unmarked word order, but violate Reis' definiteness condition (+def < −def); the remaining sentences in (62) and (63) exhibit marked word order, and (62-b) and (63-b) also violate the definiteness condition. According to Reis' judgement, the sentences that violate the definiteness condition are worse than (62-c) and (63-c), even if they exhibit the unmarked order (indirect object before direct object, cf. section 4.3.4). From this she concludes that what Lenerz calls 'the definiteness condition' cannot be used to distinguish marked and unmarked word orders.

As mentioned above morpho-syntactic definiteness is a feature of a constituent and not a contextual condition, thus it should not be used to distinguish between marked and unmarked word orders anyway. The example in (64-a) shows that the word order of (62-c) is marked for independent reasons.

(64) Was hat er einem Jungen gestohlen?
a. ??Er hat das BUCH einem Jungen gestohlen.
b. Er hat einem Jungen das BUCH gestohlen.

We cannot focus a direct object preceding an indirect object, even if the direct object is definite and the indirect object is indefinite, which speaks for Lenerz' view that definiteness does not play a role for unmarked word orders. For what concerns Reis' acceptability ratings: Also to me (62-c) and (63-c) seem marginally better than their counterparts in (62-a) and (63-a). The focus on *gestohlen* in (62) indicates a set of alternative verbs, but I find it hard to imagine a context that would prompt a speaker to focus the verb and mention the boy without an accent (if stealing the book from a boy was relevant, the constituent should receive an accent, otherwise I would expect it to be dropped), which is probably the reason why all three sentences in (62) sound a

bit strange, but in (65) the second sentence (direct object < indirect object) also seems slightly better than the first one (indirect object < direct object).

(65) Wem hat er das Buch gestohlen?

a. ?Er hat einem JUNGen das Buch gestohlen.

b. Er hat das Buch einem JUNGen gestohlen.

There are two different conclusions one can draw from examples like (63) and (65): Maybe marked word orders are sometimes preferred over unmarked word orders if their specific conditions are met by the context (but the unmarked order is still acceptable); or maybe it is not always possible to find a truly unmarked order of constituents, but only an order that is 'less marked' than all other orders, that is, which may appear in a maximal number of contexts, which do not necessarily form a superset of all the other orders' contexts. In any case, because of (64) and preliminary tests, which showed that using definiteness to model the unmarked word order lowers the performance of the system, the feature is ignored.

4.3.3. Implementation

The linearization component performs a simple comparison-based insertion: A function maps each sentence-level constituent in the parse (or more precisely, a tuple of the constituent's STTS-tag and its grammatical function) to a natural number indicating its relative position in a question with unmarked word order (see Table 4.9 for all cases; a star matches anything, vertical bars separate alternatives). The score of the prefield constituent is compared to the score of each of the remaining constituents in the sentence. The prefield constituent is inserted after the last constituent with a lower score. For an example, look at the question in (66-b) the system generated for the sentence in (66-a)[29].

(66) a. [S Sie habe immer darauf geachtet, Parteikommunikation und Regierungskommunikation zu trennen,] [VVFIN antwortete] [NE-SB Merkel] [PP-TIME am Freitagmorgen] [PP-MNR beim EU-Gipfel in Brüssel] [PP-OP auf eine entsprechende Frage].

b. Wer antwortete am Freitagmorgen beim EU-Gipfel in Brüssel auf eine entsprechende Frage, er habe immer darauf geachtet, Parteikommunikation und Regierungskommunikation zu trennen?

[29] The sentence is taken from the example input text that comes with CorZu.

Score	Feature pairs	Description
0	(VVFIN|VMFIN|VAFIN, ∗)	*left bracket:* finite verb
1	(PPER, SB), (∗, EP)	pron. subject or expletive pronoun
2	(PRF, OA)	reflexive pronoun
3	(NP|NE|NN|MPN, SB)	nominal subject
4	(PPER, OA)	acc. object personal pronoun
5	(PPER, DA)	dative personal pronoun
6	(PPER, OA2|OG)	second acc./genitive pers. pronoun
7	(AVP|ADV, ∗), (PP, TIME)	adverbial phrase or temporal PP
8	(∗, DA)	dative object/free dative
9	(∗, OA)	accusative object
10	(PTKNEG, ∗)	negation particle
11	(∗, MO), (∗, ∗)	modifier or default
12	(∗, OA2|OG|OP)	second acc./gen./prepos. object
13	(VVINF|VVPP|VAINF|VMPP |VAPP|VVIZU|PTKVZ|VP, ∗), (∗, SVP)	*right bracket:* nonfinite verb (infinitive, participle), verb particle
14	($,, ∗)	comma
15	(S, ∗)	clause
16	($., ∗)	sentence final punctuation

Table 4.9.: Mapping from feature triples to relative order scores.

The labeled brackets indicate sentence-level constituents as obtained from the Stanford parser[30]. In Table 4.9 we can look up the score for each constituent. The prefield constituent has the score 15. Because 15 is greater than 12, the score of the last constituent in (66-a), the clause is moved to the last position. The score of the prefield constituent in this case is also greater than the score of each intermediate constituent, but this need not be the case, as it is inserted after the last constituent with lower score.

4.3.4. Transitive Rules

The following transitive rules describe the unmarked order of different constituents after the prefield. Numbers in brackets refer to the corresponding order scores in Table 4.9. They should appear in non-decreasing order from left to right, otherwise the system will generate questions with marked word orders. If a rule is simply taken over from the literature, I do not repeat the whole argument[31] but refer to the source.

[30]For the sake of simplicity, I ignore the comma, which also is a sentence-level constituent in the parse and needs to be moved.

[31]To show that a word order is unmarked, one needs to show that this order is not subject to the contextual constraints that restrict the occurrence of alternative word orders (see Lenerz' definition above) which requires a lot of examples and space.

Objects

Among nominal phrases, indirect objects (IO) precede direct objects (DO) (Lenerz, 1977, pp. 39–63):

(67) Full nominal phrases
IO [8] < DO [9]

(68) Der Postbote gibt der Frau einen Brief.

This rule covers the overwhelming majority but not all of the cases involving indirect and direct object NPs. Examples like (69), taken from Frey and Pittner (1998, p. 497), seem to suggest that the unmarked word order is underlyingly determined by theta roles, not grammatical functions.

(69) a. Man hat das Auto dieser Prüfung noch nie unterzogen.
b. ??Man hat dieser Prüfung das Auto noch nie unterzogen.
c. Dieser Prüfung unterzogen hat man das Auto noch nie.
d. ??Das Auto unterzogen hat man dieser Prüfung nie.

Grammatical functions and theta roles assigned by different verbs to their arguments are not always aligned in the same way. The theta role of the dative argument of *unterziehen* is neither an *affected object* nor a *benefactive* according to the roles defined by Polenz (2008, pp. 170f.), but rather an *instrument* that is part of the predicate. Example (69-c) shows that the verb and the dative argument form a constituent that can be topicalized, which indicates that the dative object of *unterziehen* is base-generated lower than the accusative object. For the majority of verbs (e.g., *geben*, which a assigns a benefactive role to its dative argument) it is exactly the other way round, which would explain the different linearizations.

Among personal pronouns, direct objects precede indirect objects (Lenerz, 1977, pp. 68f.). Hofmann (1994, pp. 49ff.) furthermore identifies a set of rules for different subclasses of pronouns. She distinguishes deictic, substituting, demonstrative and indefinite pronouns. Substituting pronouns fall under the rule in (70). Questions with deictic or demonstrative pronouns are unspecific, thus it is not worth modeling their linearization. Indefinite pronouns, as far as I can see, behave similarly to full nominal phrases, so they are covered by the rule in (67).

(70) Personal pronouns
DO [4] < IO [5]

(71) Der Postbote gibt ihn ihr.

Objects that are personal pronouns (PPER-O) precede objects that are nominal phrases (NP-O). Subjects (SU) precede objects that are personal pronouns (Hofmann, 1994, pp. 64–68) and, by transitivity, all other objects:

(72) Personal pronouns and full nominal phrases

a. PPER-O [4, 5] < NP-O [8, 9]

b. SU [1, 3] < PPER-O [4]

(73) a. Der Postbote gibt ihr den Brief.

b. Am Montag gab der Postbote ihr den Brief.

For some verbs the unmarked order is reversed, e.g., the objects of psych-verbs and verbs like *gelingen*, *gehören* and *zustehen* precede the subject, see Lenerz (1977, pp. 114–116) for an incomplete list of verbs and example sentences. This seems to be further evidence for the theta-role hypothesis, since the subjects of these verbs mostly are not assigned the proto-agens role that is most common among other subjects. Currently the system does not cover this exception because it cannot reliably identify these verbs.

Nominal phrases that are direct objects precede prepositional objects (PO) (Lenerz, 1977, p. 65–68). This holds independent of the type of the nominal phrase governed by the head of the prepositional phrase.

(74) Nominal phrases and prepositional objects
DO [9] < PO [12]

(75) Jemand hat den Brief an die Frau adressiert.

Second accusative objects and genitive objects appear in the same position as prepositional objects (and no two of them ever appear together in the same verb valency). Any rule in this section that applies to prepositional objects also tacitly applies to second accusative and genitive objects.

Adverbials

The distribution of adverbials is very complex: In their detailed analysis of adverbial positions in the German midfield, Frey and Pittner (1998) identify five broad subgroups of adverbials that are generated in different syntactic base positions. Syntactic base positions need not necessarily be reflected in the linear order at the surface[32], but Frey and Pittner (1998) make very clear predictions about observable word orders based on

[32]Dependening on the syntactic theory, the relation between theoretical structural positions and empirically observable data can be opaque due to syntactic operations that apply after base generation.

their structural analyses, which they back up with various tests[33], allowing me to use their results as a basis for modeling the unmarked order of adverbials. In the following, I present the five subgroups in the order in which they appear in Frey and Pittner (1998), from right to left, that is, from the adverbials with the narrowest scope to those with the widest scope.

The first subgroup are *process-related* adverbials, which immediately modify the predicate, e.g., manner adverbials and directional adverbials. The predicate may be simple (only the verb (76-a)) or complex (including an object (76-b)), if the object can be integrated in the sense of Jacobs (1993, 1999). Process-related adverbials immediately precede the predicate complex. With the example in (76-c), Frey and Pittner (1998) show that adverbials that otherwise may have a process-related reading can even have a different meaning in a higher position (that is, if they appear before the object).

(76) a. Er muss das Geschirr langsam spülen.
b. weil Hans sorgfältig ein Hemd bügelte
c. Er muss langsam das Geschirr spülen.

This subgroup corresponds to the third of the three groups of adverbials that Hoberg (1977) identifies by looking at the relative positions of adverbials with respect to the negation[34] and the verb that fills the right bracket in a corpus of 11,000 sentences. Her finding that adverbials of this group always follow the negation is in line with the linearization rule above, if the negation is not seen as part of the predicate complex.

Adverbials of the second subgroup are *event-internal* (or *situation-internal* (Pittner & Berman, 2007, p. 151)). Event-internal adverbials can be instrumental (77-a), comitative (77-b) and local adverbials (77-c) as well as adverbials that characterize the attitude of the subject's (or highest argument's) referent towards an event (77-d). Frey and Pittner (1998) argue that their base position is between the highest argument and the direct object.

(77) a. weil Peter mit einem Besen den Fußweg kehrt.
b. weil Peter mit seinem Freund den Fußweg kehrt.
c. weil Peter draußen den Fußweg kehrt
d. weil Peter gerne den Fußweg kehrt.

[33]These include tests for possible focus projections, themes and rhemes in the sense of Lenerz (1977), syntactic and semantic scopes, complex prefields, the position of adverbials with respect to indefinite w-pronouns and principle C effects.

[34]'Negation' here only refers to sentence negation (total negation) with the particle *nicht*, not to other negation words, which may be adverbials, determiners, indefinite pronouns or what the STTS calls 'answer particle' (*nein*), or *nicht* as word or phrase negation (partial negation). Since the system cannot distinguish between partial and total negation, there might be some problems.

This is not in line with Lenerz' (1977) unmarked order of local adverbials and objects. He first notes that the unmarked order of temporal (TEMP) adverbials and local adverbials (LOC) in German is the opposite of their unmarked order in English (Lenerz, 1977, pp. 78–82)[35]:

(78) TEMP [7] < LOC [11]

This corresponds to Frey and Pittner's (1998) distinction between event-internal and event-related adverbials (see below), but for sentences with one object, indirect (IO) or direct (DO), Lenerz (1977, pp. 85–89) finds that local adverbials must follow the object:

(79) IO [8], DO [9] < LOC [11]

His argument is that local adverbials before a single object (Frey and Pittner's (1998) base position for event-internal adverbials) cannot have narrow focus on them (Lenerz, 1977, p. 86)[36]:

(80) Wo hast du meine Frau gesehen?

a. Ich habe deine Frau in BerLIN gesehen.

b. *Ich habe in BerLIN deine Frau gesehen.

Frey and Pittner's (1998) argument[37] for their base position is the exact opposite: They say that sentences with event-internal adverbials following the object cannot have a maximal focus, which they take as evidence for a movement of the event-internal adverbial out of its base position[38]. I will not go into further details here because my intuitions are not very clear on this issue. Since the system cannot distinguish between process-related and event-internal adverbials, I adopted Lenerz' unmarked order, which is the same for both. To describe the unmarked order of temporal and local adverbials and multiple objects, Lenerz (1977, pp. 87–89) splits local adverbials further into two subgroups: Obligatory local adverbials appear between the direct object and the prepositional object (81). Facultative local adverbials appear between the temporal adverbial and the indirect object (82).

(81) Obligatory local adverbials (process-related)

[35]However, cf. Frey and Pittner (1999) for a comparison of adverbial positions in German and English which concludes that the regularities identified for the different subclasses of adverbials also hold for English and that apparent differences in the linearization must be due to more general syntactic constraints.

[36]The same holds for instrumental and comitative adverbials.

[37]Actually, they also have evidence from indefinite w-pronouns, scopus tests and complex prefields, which I ignore here, since I am interested in the unmarked order with respect to information-structural properties of the context, see above.

[38]The trace of the adverbial between the verb and the object blocks focus projection.

TEMP [7] < IO [8] < DO [9] < LOC [11] < PO [12]

(82) Facultative local adverbials (event-internal)
TEMP [7] < LOC [11] < IO [8] < DO [9]

In Lenerz' examples, obligatory local adverbials always have directional semantics, which means they are process-related according to Frey and Pittner (1998), whereas facultative local adverbials are never directional and belong to the event-internal adverbials. Thus, in the context of multiple objects, both analyses make the same predictions. The system, however, only models the unmarked order of obligatory (or process-related) adverbials, as can be seen from the unsorted numbers in brackets. All the above rules for event-internal adverbials do not apply to pronominal objects, which always precede these adverbials (Lenerz, 1977, p. 89):

(83) Ich habe es ihm damals gegeben.

Adverbials of the third subgroup are *event-related* (or *situation-related* (Pittner & Berman, 2007, p. 151)) and can have temporal, causal or habitual semantics. Event-related adverbials refer to the event specified by the sentence, hence their base position needs c-command over the base positions of all arguments. Frey and Pittner (1998) predict that they should precede the highest argument. That these adverbials are predicted to precede the objects as well as process-related and event-internal adverbials is consistent with Lenerz' orders in (81) and (82). However, according to Lenerz, these adverbials in the unmarked order do not precede the subject (see below). Frey and Pittner (1998) justify their assumption with the position of w-indefinites, which are usually said to stay in their base position (i.e., they do not scramble):

(84) a. ??weil wer morgen den Balken abstützen sollte
b. weil morgen wer den Balken abstützen sollte

Example (84-a) is not completely unacceptable, but (84-b) seems better. Again, there is a problem with the narrow focus on the adverbial if it precedes the subject, see example (85), copied from Lenerz (1977, p. 98), and maybe a problem with the maximal focus if the subject precedes the adverbial (86).

(85) Wann ist der Posträuber ausgebrochen?
a. Ich glaube, dass der Posträuber GEStern ausgebrochen ist.
b. *Ich glaube, dass GEStern der Posträuber ausgebrochen ist.

(86) Was ist passiert?
a. ?Ich glaube, dass der Posträuber gestern ausgebrochen ist.

b. Ich glaube, dass gestern der Posträuber ausgebrochen ist.

As I already said above, my intuitions on these examples are not very clear. I would say that a focus on the whole embedded clause in (86-a) is possible with an accent on the subject, but Frey and Pittner (1998) might object. In any case, (85-b) is clearly worse than (86-a), so I will adopt Lenerz' unmarked word order for practical reasons.

Adverbials of the fourth subgroup are called *sentence adverbials*. To these belong, for example, the adverbs *wohl*, *allerdings*, *vielleicht* and *tatsächlich*. They refer to the proposition of the whole sentence and hence need scope over the base positions of event-related adverbials and the finite verb. This subgroup corresponds to the first group of adverbials identified by Hoberg (1977). She finds that sentence adverbials always precede negation, which is what we expect if they are to modify the whole proposition.

(87) S-ADV [7, 11] < NEG [10]

The system does not recognize sentence adverbials (they appear as immediate children of the sentence in the constituency parse and are annotated as modifiers of the verb by the dependency parser), thus they may either get the order score 7 (the same as for temporal adverbials) if they appear as adverbs or adverbial phrases or the score 11 if the dependency parser analyzes them as modifiers. We could try to define an exhaustive list of adverbs that can appear as sentence adverbials, but sentence adverbials need not always be adverbs, they can, for example, come as prepositional phrases or participle phrases.

The highest syntactic position among adverbials is occupied by *frame adverbials* and *pragmatic* or *speech-act* adverbials. Frame adverbials provide a frame according to which the truth value of a proposition is evaluated, see (88-a) taken from Frey and Pittner (1998, p. 518). Speech-act adverbials comment on certain aspects of an utterance and have parenthetical character like the adverbial in Joschka Fischer's famous quote in (88-b) (Pittner & Berman, 2007, p. 150).

(88) a. weil im Mittelalter erstaunlicherweise die Mönche während der Fastenzeit viel Bier tranken
b. Mit Verlaub, Sie sind ein Arschloch.

For both kinds of adverbials we have the same problem as with sentence adverbials: They cannot be identified by the system. Additional problems may arise for frame adverbials in initial position because the system expects that in V2 sentences the position before the finite verb is occupied by only one constituent.

Subjects

Subjects occupy the position before the temporal adverbial and the objects (Lenerz, 1977, pp. 97–106).

(89) Subjects precede temporal adverbials
SU [1, 3] < TEMP [7]

Hofmann (1994, p. 66f.) adds an exception for objects that are expressed by deictic pronouns, which, according to her, may precede the subject in the unmarked order. As evidence, she gives the two examples in (90).

(90) a. weil mir die Oma den Macho vorstellte
b. weil mich die Oma der Feministin vorstellte

(91) a. Die Oma stellte mir den Macho vor.
b. Die Oma stellte mich der Feministin vor.

But if we imagine that the hearer did not understand the deictic pronoun in one of the sentences in (91) and hence asks for the indirect object (92) or the direct object (93), we can see that the deictic pronouns cannot be focused when they precede the subject.

(92) Wem stellte die Oma den Macho vor?

a. ??Ich habe gesagt, dass MIR die Oma den Macho vorstellte.
b. Ich habe gesagt, dass die Oma MIR den Macho vorstellte.

(93) Wen stellte die Oma der Feministin vor?

a. ??Ich habe gesagt, dass MICH die Oma der Feministin vorstellte.
b. Ich habe gesagt, dass die Oma MICH der Feministin vorstellte.

Thus, the word order in (90) is not unmarked according to Lenerz' definition. I think what Hofmann actually tried to capture is the unmarked order of reflexive pronouns. Reflexive pronouns can have the same form as the deictic pronouns above, but they have no independent grammatical function. They belong to the verb and usually cannot be focused independently, so they are not subject to the constraint that ruled out deictic pronouns preceding subjects in (92) and (93). Whenever a nominal subject (N-SU) has to be moved out of the prefield and there is a reflexive pronoun (PRF) in the midfield, the system will move the subject behind the reflexive pronoun (95-a), although the reverse order would be perfectly fine as well (95-b).

(94) Reflexive pronouns precede nominal subjects
PRF [2] < N-SU [3]

(95) a. Gestern hat sich seine Frau uns vorgestellt.
b. Gestern hat seine Frau sich uns vorgestellt.

Subordinate Clauses

Sentence-level[39] subordinate clauses are usually extraposed to the postfield in questions.

(96) Subordinate clauses
RB [13] < S [15]

(97) Wann hat sie ihm erzählt, dass sie schwanger ist?

There are some exceptions, see the questions in (98). In this case, the clause is not extraposed, but seems to occupy the subject position.

(98) Wer den Armen helfen will, spendet der Caritas Geld.
a. Was spendet, wer den Armen helfen will, der Caritas?
b. ?Was spendet der Caritas, wer den Armen helfen will?
c. Wem spendet, wer den Armen helfen will, Geld.
d. ?Wem spendet Geld, wer den Armen helfen will.

4.4. Resolving Coreferences

To avoid vague questions, deictic expressions, pro-forms whose antecedents are not contained in the question itself and semantically underspecified nouns whose referent is only identifiable in a certain context must be resolved. Deictic expressions include, for example, first and second person personal pronouns or temporal and local adverbs like *gestern* and *hier*. Pro-forms are different kinds of pronouns and pronominal adverbs. Currently, the system only resolves personal pronouns. As described in section 3.7, each node in the constituency tree deriving a personal pronoun should already be annotated with the subtree of its antecedent, given the coreference resolution did not fail. So, if a pronoun is not bound within the question, we only need to replace it with its antecedent and adjust the inflection of the antecedent according to the morpho-syntactic features of the pronoun.

To find the personal pronouns that need to be resolved, a postorder traversal is performed on the constituency tree of the question. Each personal pronoun for which no coreferring element was encountered before is replaced by its antecedent.

A special case occurs if a personal pronoun has the same referent as the interrogative pronoun, as in the question generated for the sentence in (66), repeated in (99).

[39] This excludes relative clauses.

(99) a. Sie habe immer darauf geachtet, Parteikommunikation und Regierungskommunikation zu trennen, antwortete Merkel am Freitagmorgen beim EU-Gipfel in Brüssel auf eine entsprechende Frage.

b. Wer antwortete am Freitagmorgen beim EU-Gipfel in Brüssel auf eine entsprechende Frage, er habe immer darauf geachtet, Parteikommunikation und Regierungskommunikation zu trennen?

Replacing the pronoun with *Merkel* would implicate that the persons asked for by the question and referred to by *Merkel* are not the same. Thus, the system needs to remember and take into account the referent of the phrase that is replaced by the question phrase.

The question in (99-b) also illustrates another problem: Traditionally it is assumed that the inherent gender of *wer* is masculine. But as the generic nominative interrogative pronoun for persons, it can elicit noun phrases of any gender (e.g., *die Frau*, *der Mann*, *das Mädchen*). So, if the system replaces a feminine noun phrase with *wer*, *wem* or *wen* and there are pronouns in the sentence referring to this noun phrase, these pronouns need to be replaced by their generic masculine forms (this is what happened in question (99-b)). However, it is not always as simple as that: In a blog post[40] Anatol Stefanowitsch collected some interesting examples that seem to suggest that *wer* in certain contexts and/or for certain speakers can be feminine (or that the agreement in this case is not sensitive to the morpho-syntactic gender feature but the semantic sexus feature of the noun phrase). In (100), I copied some of his examples.

(100) a. Wer von euch ist schwanger und hatte trotzdem seine Periode?

b. Wer von euch hat ihre Tage auch erst ganz spät bekommen und hat nie starke Blutung?

c. Wer von euch kann mir seine/ihre Erfahrungen – nur zu diesen beiden Reifen – mitteilen.

The first two questions clearly refer exclusively to women, yet the first one uses a generic masculine pronoun while the second one keeps the feminine pronoun. Both examples stem from the same forum for women. The question in (100-c) has both forms (apparently to explicitly address both genders) and was written by a male user in a forum for cars and motor bikes. Stefanowitsch mentions two possible explanations for these heterogeneous examples: Language change or synchronous varieties. I do not concern myself here with the explanation of the phenomenon, but simply decide to stick to the generic forms. If a user of the question generation system prefers gender-specific forms, the method adjusting pronouns to generic forms just needs to be commented out.

[40] See `http://www.sprachlog.de/2013/04/18/wer-ist-maskulin-wer-ist-feminin/`.

Questions like (101) suggest that not just pronouns, but also nouns referring to persons should be replaced by their generic forms.

(101) Wer hat sich bereits 2005, als Kanzlerin, für eine Parteifinanzierung seines Handys entschieden?

However, I think that many people do not perceive the masculine form as a generic form in cases like that (anymore?) and sometimes there simply is no generic form (e.g., for *Krankenschwester* and *Krankenpfleger*).

4.5. Inflection

The system needs a component that inflects German words. When asking for a plural or non-third person pronoun subject, the verb needs to be conjugated to match the third person singular interrogative pronoun. If the grammatical case of an antecedent does not match the grammatical case of its anaphor, determiners, adjectives, nouns and pronouns might need to be declined.

4.5.1. Reverse Lemmatizer

The lemmatizer of the RFTagger uses a lexicon that maps word forms and morphological tags to lemmas. Based on this lexicon, I try to map lemmas and morphological tags to word forms in what could be called a 'reverse lemmatizer'. The lemmatizer lexicon was, of course, not designed for this, so the reverse mapping is not a well-defined function. For one lemma-tag pair there are often several different word forms. To find the best among these forms, I ran the RFTagger and its lemmatizer on the monolingual versions of the Europarl corpus (version 7), the news commentary and the news crawl corpora from 2009 and 2010 provided for the shared task of the eighth workshop on statistical machine translation[41] and determined the most frequent word form for each lemma-tag pair. The first problem with this approach is that not all forms can be found in these corpora. However, many forms are just spelling variants: Words containing the five graphemes *ä*, *ö*, *ü* and *ß*, or their replacements *ae*, *oe*, *ue* and *ss*, words in lowercase or uppercase letters. For the lemma-tag pairs that were not seen at least once, we can simply pick the alternative with the best spelling (words with proper umlauts and *ß* are preferred over words with replacements, nouns should start with an uppercase letter and continue with lowercase letters, all other words should not contain any uppercase letters). A real problem are the different inflection patterns of German adjectives in attributive position. Attributive adjectives can be declined according to a weak, a mixed

[41] All corpora are available here: `http://www.statmt.org/wmt13/translation-task.html`.

and a strong pattern. The choice for a pattern is context-dependent. The morphological tags of the RFTagger do not give any information on the inflection class of the adjective, nor does any morphological tool that I know of. So far, this is an unsolved problem. The system in these cases returns the most frequent form or the first of several equally frequent forms.

4.5.2. Inflecting Verbs, Antecedents and Possessive Pronouns

To retrieve an inflected form from the reverse lemmatizer, a lemma and a morphological tag need to be provided. For verbs, we take the lemma of the form that needs to be conjugated: The fifth field of its morphological tag is changed to 'indicative', if the answer phrase is the subject, also the third and the fourth field are adjusted (to 'third person' and 'singular'). The general change from subjunctive to indicative forms is problematic, since the irrealis often is necessary (e.g., to indicate indirect speech), but it at least ensures that the generated question will be grammatical.

Antecedents can be complex, for example, they may contain a relative clause. To avoid inflecting words from embedded clauses, only immediate children of the antecedent's root node are inflected. For each part of speech, the position of the grammatical case feature needs to be identified to adjust it to the grammatical case of the pronoun.

Finding generic (masculine) possessive pronouns is a special case, since they encode two different genders. The gender of their antecedent, which we want to change to masculine, and (via the adjective inflection) the gender of the noun they modify, which is supposed to stay the same, see the examples in (102).

(102) a. Sie ruft ihre Freundin/ihren Freund an.
b. Er ruft seine Freundin/seinen Freund an.

In the lemma lexicon of the RFTagger, the latter distinction is indicated by a gender feature, whereas the forms in (102-b) are distinguished from those in (102-a) by a different lemma. Thus, to replace feminine by generic masculine possessive pronouns, we cannot use their annotated lemma, but always have to use *seine*.

4.6. Post-Processing

After the main linguistic work is done, some post-processing steps are performed to ensure nicely formatted output. The first character is converted to uppercase. If the question ends with a period (which was carried over from the input sentence), it is replaced by a question mark. If there is no period, a subtree with a question mark is added under the root node of the sentence. Sometimes moving the prefield constituent leads to a comma immediately preceding sentence final punctuation or another comma. These superfluous commas are removed.

5. Evaluation

The evaluation mainly consists of a qualitative analysis of questions and errors. I also report some quantitative data based on the frequency of quality scores and error types indicating the precision of the system and the relative importance of different errors. The recall of the system was not evaluated because it is difficult to define a set of questions that should have been generated for a given text: We could only consider questions for NPs, PPs and embedded clauses and the recall value would be high. Defining a set of questions that additionally ask for presupposed or inferred information is not a trivial task and the recall evaluated on such a set is expected to be rather low. Both values would not be very informative – in the first case, we artificially reduced the set of relevant questions, in the second case, we included questions that we know cannot be generated for systematic reasons. To compute precision, we do not need to know the complete set of relevant questions – we only need to be able to recognize a good question when we see it.

5.1. Data

The evaluation was performed on three newswire texts: a text from the "Ausland" (world) section with 29 sentences, a text from the "Inland" (Germany) section with 28 sentences and a text from the "Unternehmen" (business) section with 36 sentences[1]. The selection is not random, as I tried to find longer texts of similar length with a variety of linguistic phenomena, e.g., nominal and prepositional phrases with different grammatical functions and semantics, personal pronouns and subordinate clauses with and without correlates. Neswire texts were chosen as source because they are not trivial from a linguistic point of view but also not too complex for the parser. Intermediate tests showed that the performance of the constituency parser tends to be much lower on Wikipedia texts. In order to use the system on sentences of this complexity, we would need a text simplification component to extract simpler sentences from the input, see section 5.4.

[1] "86 Tote bei Doppelanschlag auf Friedensmarsch in Ankara" (October 10, 2015), "Bundestag beschließt umstrittene Vorratsdatenspeicherung" (October 16, 2015) and "Abgasskandal und China-Schwäche perlen an Daimler ab" (October 22, 2015), all three texts were taken from the main page of Reuters Germany. All examples in this section are directly taken from or generated based on these texts.

5.2. Results

The system generated questions for 77 of the 93 sentences in the three texts. Table 5.1 shows the number of questions generated for different types of answer phrases, where an answer type is a combination of grammatical category and function. The functions were corrected manually. If the answer phrase does not form a constituent and thus does not have a grammatical function, it is marked as fragment (FRAGM). The syntactic categories were taken directly from the constituency parse. In total, 150 questions were

	World	Germany	Business	Total
CNP-SB	1	2	0	3
MPN-SB	0	0	1	1
NE-DA	0	0	1	1
NE-OA	0	0	1	1
NE-SB	0	3	3	6
NN-SB	1	1	0	2
NP-DA	1	1	0	2
NP-FRAGM	1	0	0	1
NP-MO	1	0	0	1
NP-OA	8	7	3	18
NP-SB	17	19	16	52
PP-CVC	0	1	0	1
PP-FRAGM	3	2	1	6
PP-MNR	3	1	2	6
PP-MO	9	7	14	30
PP-OP	4	3	2	9
PP-PG	0	1	0	1
PP-SBP	1	0	0	1
S-MO	0	0	2	2
S-OC	0	4	1	5
S-SB	0	0	1	1
Total	50	52	48	150

Table 5.1.: Number of questions per answer type (category and function).

generated, that is, almost two per sentence on average (excluding the sentences for which no questions were generated at all). For each text, we have roughly the same number of questions. Since the third text has more sentences than both the first and the second text, this means that on average fewer questions were generated from the sentences of the third text. The most frequent answer phrase types are subjects, PP modifiers and accusative objects. This does not come as a surprise, since the system tries to generate questions for all syntactically possible targets and these three types are

Score	Description
1	ungrammatical
2	grammatical
3	acceptable without context
4	acceptable given the source sentence
5	specific enough (not vague)
6	interesting (answer is informative), not too complex

Table 5.2.: Question quality scores, ordered from worst to best.

simply the most frequent groups (almost each sentence has a subject[2], PP modifiers may appear multiple times in one sentence, and accusative objects frequently appear with transitive verbs). Unfortunately, the number of embedded clauses in the three texts is quite low, so we will not be able to say much about the generation of questions that target different kinds of subordinates[3]. Apart from the seven fragments, Table 5.1 also shows two other unexpected answer types: a pseudo-genitive (PG) and six postnominal modifiers (MNR). In these cases, the analysis of the dependency parser is correct, the prepositional phrases are attached to an NP, not to the VP. However, the constituency parser has put the PPs directly under the S node, thus the movement restrictions did not apply and the system falsely identified them as target phrases.

5.2.1. Quality Rating and Characteristic Examples

The quality of a generated question was rated according to the six-point scale in Table 5.2. As with the traditional judgements about *grammaticality* and *acceptability* in theoretical linguistics, there is an implicature from higher to lower (positive) categories. For example, if a question is classified as vague, grammaticality and acceptability are implied. In the evaluation, each question is assigned only the highest possible score. In the following, I will define each question quality level more precisely and give concrete examples of questions generated by the system in the evaluation.

The first level corresponds to the notion of ungrammaticality in theoretical linguistics. Questions of this category violate basic grammatical rules, for example, rules about syntactic arguments (1-a) or subject agreement on the finite verb (1-b).

(1) Score 1: ungrammatical

a. *Wo warnte die Grünen-Rechtspolitikerin Renate Künast der Daten?

b. *Wem sagte zwei Insider, die Behörden gingen Berichten über einen Selbst-

[2]Although some only have dummy subjects which are no good targets.

[3]Of course, one reason for using newswire text was that it is less complex, as mentioned above. So this does not really come as a surprise as well.

mordanschlag nach?

The number and nature of the errors that led to a question's ungrammaticality are irrelevant. Thus, this rating scheme is a bit stricter than the one that was used in the QGSTEC'10 (cf. section 2.2), according to which question (1-b) probably would have received a higher score than question (1-a).

The second-lowest score is given to questions that are grammatical, but unacceptable. The problems that render a question unacceptable are mostly semantic in nature. The most common error is a question phrase that is incompatible with the semantic role assigned by the verb (2-a). Another instance of this category are unidiomatic questions like (2-b).

(2) Score 2: Unacceptable

a. ??Wer fand zu einer Zeit statt, in der die türkischen Sicherheitskräfte gegen Kurden- und Islamisten-Gruppen vorgehen?

b. ?Was sprach Bundeskanzlerin Angela Merkel in einem Schreiben an Ministerpräsident Ahmet Davutoglu aus?

Up to this point all judgements are based entirely on the question. To distinguish between score three and four, the question must be compared to the source sentence (the sentence from which it was generated) and its context.

The third level is for questions that are acceptable but not faithful to the semantics of the source sentence. At first sight, these questions may look very good, but they usually ask for information that is not given in the source sentence. Common patterns are false PP attachments like in question (3-b) – in the source sentence (3-a), the PP is attached low (to the NP), but the system falsely attaches it to the VP – and interchanged grammatical functions due to marked word orders and case syncretisms (3-d).

(3) Score 3: Acceptable in isolation

a. Die Demonstranten forderten ein Ende des Konflikts zwischen türkischem Militär und Kurden im Südosten des Landes.

b. Wo forderten die Demonstranten ein Ende des Konflikts zwischen türkischem Militär und Kurden? – im Südosten des Landes

c. Die Zahlen für das dritte Quartal bewerteten viele Fachleute positiv.

d. Wer bewertete viele Fachleute positiv? – die Zahlen für das dritte Quartal

Questions with score four are grammatical and acceptable (both in isolation and with respect to the source sentence), but they do not allow for a clear answer because they are vague. Reasons for vagueness can be unresolved pro-forms (4-b) and underspecified noun phrases, predicates or temporal and local expressions (or a combination of these

factors like in (4-d), where both the predicate and the temporal expression contribute to the vagueness of the question).

(4) Score 4: Acceptable given the source sentence

a. Die türkische Regierung lässt zudem Stellungen der Extremisten-Miliz Islamischer Staat (IS) in Syrien bombardieren.
b. Wer lässt zudem Stellungen der Extremisten-Miliz Islamischer Staat (IS) in Syrien bombardieren? – die Türkische Regierung
c. Bereits vor einigen Tagen hatte es Hinweise auf diese Entscheidung gegeben.
d. Was hatte es bereits vor einigen Tagen gegeben? – Hinweise auf diese Entscheidung

Under optimal conditions, that is, if the automatic annotation was one hundert percent correct all the time, all generated questions should at least achieve a score of four.

The next level requires questions to be specific, which means that an attentive reader of the text containing the source sentence should be able to clearly answer the question. However, these questions are still not ideal, since they are either to complex or the requested information has only little value, for example, the answers for both (5-a) and (5-b) can almost be guessed without any knowledge from the text, and (5-a) additionally is quite complex.

(5) Score 5: Specific enough

a. Worauf kann sich Daimler voll konzentrieren, während VW bei Millionen Autos Software und Motortechnik erneuern muss, und sich auch die Tochter Audi mit den Folgen aus dem Diesel-Skandal samt zahlreicher Führungswechsel im Konzern herumschlagen muss? – aufs Geschäft
b. Worin schrieb Analyst Frank Schwope von der NordLB, bei Daimler dürften sich „die nächsten zwei Geschäftsjahre als relativ ertragreich darstellen“? – in einer Analyse

The highest score is reserved for optimal output, that is, specific and informative questions that are not unnecessarily complex:

(6) Score 6: Interesting question that is not too complex

a. Womit hatten die Wolfsburger Abgas-Tests von Millionen Diesel-Fahrzeugen manipuliert, um sie klimafreundlicher erscheinen zu lassen? – mit einer speziellen Software
b. Wer will bis 2020 BMW als führenden Premiumhersteller der Welt ablösen? – die Stuttgarter

c. Wann hatte der EuGH eine geplante EU-Regelung zur Speicherung von Daten ohne Verdacht auf Straftaten für nichtig erklärt? – vor einem Jahr

5.2.2. Quantitative Results

As said above, the samples selected for the evaluation contain only relatively few sentences and the selection was not completely random, so we should not overinterpret the quantitative results. Nevertheless, the numbers give us a first overview and may reveal some broad tendencies.

Table 5.3 shows the relative frequencies of questions with different quality scores generated from each text and in total[4]. We can see that the distribution of quality scores

	World	Germany	Business	Total
1	0.06	0.12	0.12	0.10
2	0.14	0.13	0.23	0.17
3	0.16	0.15	0.08	0.13
4	0.10	0.08	0.17	0.11
5	0.04	0.04	0.06	0.05
6	0.50	0.48	0.33	0.44

Table 5.3.: Relative frequencies of quality scores (1–6).

is very similar across the three texts. The largest share of questions has the highest score and 65–80% are at least acceptable. The results for the third text (business news) are slightly worse than for the other two, as more unacceptable and vague questions were generated. This is in line with the fact that on average also fewer questions per sentence were generated for this text (cf. Table 5.1).

For most types of answer phrases there are not enough data to say something about the type-specific distribution of quality scores, but for subjects and PP modifiers, the distribution is very similar (roughly 52% of the subjects and 46% of the PP modifiers received the highest score). In Table 5.4, we see the distribution of quality scores among questions with different question phrases. Given the small size of the input, the system generated quite a variety of different question phrases. For most of them, we only have very sparse data, so we do not know how reliably they can be generated. In the majority of cases, there is at least one acceptable question. *Was* questions are the default for subjects, NP objects and embedded clauses – that means, any false or missing annotation (and, of course, any false linguistic generalization) may lead to an unacceptable *was* question, which explains the high number of low scores (cf. the error analysis below).

If we look at different topological fields, we can see that the probability of generating a question of the highest quality is more than ten percent higher for answer phrases

[4]Columns should add up to one, except for rounding errors.

Question phrase	1	2	3	4	5	6	Total
Was	8	14	5	6	2	14	49
Wer	0	4	4	4	0	28	40
Wo	2	2	2	1	0	5	12
Wann	1	1	0	2	0	1	5
Wie lange	0	1	1	0	0	2	4
Wohin	1	1	1	0	0	1	4
Worin	0	1	0	0	2	1	4
Wovon	0	0	3	0	0	1	4
Wobei	0	0	0	0	1	2	3
Wonach	0	0	1	1	0	1	3
Worauf	0	0	0	0	1	2	3
Wofür	0	0	0	1	0	2	3
Womit	0	1	0	0	0	2	3
Seit wann	0	0	0	0	0	2	2
Wem	2	0	0	0	0	0	2
Von wem	0	0	1	0	0	0	1
Woran	0	0	0	1	0	0	1
Wozu	0	0	0	0	0	1	1
Zwischen wem	0	0	0	0	0	1	1
Ohne was	0	0	1	0	0	0	1
Bei wem	0	0	0	1	0	0	1
Um was	0	0	0	0	1	0	1
Warum	0	0	0	1	0	0	1
Woraus	1	0	0	0	0	0	1

Table 5.4.: Absolute frequencies of quality scores and question phrases.

from the prefield than for answer phrases from the midfield (for the postfield, we have only very sparse data). This is due to the fact that even complex phrases are usually recognized correctly in the prefield, whereas parser errors are frequent in the midfield because of attachment ambiguities.

5.2.3. Error Analysis

Six parses contained a non-unary root (NUR) category (see section 3.2) – for the sentences with these parses, no questions could be generated. However, these errors do not appear in the following, as I only discuss problems with generated questions.

Table 5.5 lists all types of errors that were identified as causes for suboptimal questions. The first part of the list refers to errors in the automatic linguistic annotation. The two points in the middle concern the core of the system and should have been solved already. The last three problems eventually need to be be solved by a good QG system, but have not been addressed so far: The system overgenerates for certain types of embedded

clauses, it does not resolve (or filter out) pro-forms other than personal pronouns and it does not detect expressions with underspecified semantics. There is no code for errors

Error code	**Description**
CP	constituency parse
CR	coreference resolution
DP	dependency parse
GF	grammatical function
MA	morphological analysis
SCN	semantic category (noun)
SCV	semantic category (verb)
AQ	question-answer phrase mapping
LO	linear order
OG	overgeneration
PR	unresolved pro-forms and conjunctive adverbs
US	underspecified semantics (out of context)

Table 5.5.: Codes for different types of errors.

caused by the inflection model, as no such errors occurred in the evaluation. I manually annotated each question that received a score between one and four with one or more error types. I did not annotate all errors, but only those that would need to be fixed for the question to receive a quality score of five or six (e.g., sometimes errors in the morphological analysis can be recovered by using the grammatical function instead, cf. section 4.2.1).

Figure 5.1 shows the absolute frequencies of the error types from Table 5.5 in the three selected texts. By far the largest number of low-quality questions is caused by errors in the constituency parse. These errors can be very different in nature, some of them are hard to avoid, for others improvements seem possible. A common error that belongs to the first category are PP attachment problems as in (3-a). On a purely syntactic level, these cases are often ambiguous. To resolve these ambiguities, additional information about the semantics and the context of the utterance and maybe even world knowledge may be necessary. Unfortunately, PP attachment ambiguities account for the largest share of parser-related problems. Another problematic case are disjunct answer phrases:

(7) a. Er fand zu einer Zeit statt, in der die türkischen Sicherheitskräfte gegen Kurden- und Islamisten-Gruppen vorgehen.
b. Wann fand der Angriff statt, in der die türkischen Sicherheitskräfte gegen Kurden- und Islamisten-Gruppen vorgehen? – zu einer Zeit

In (7-a), the relative clause appears extraposed to the postfield and the verb particle

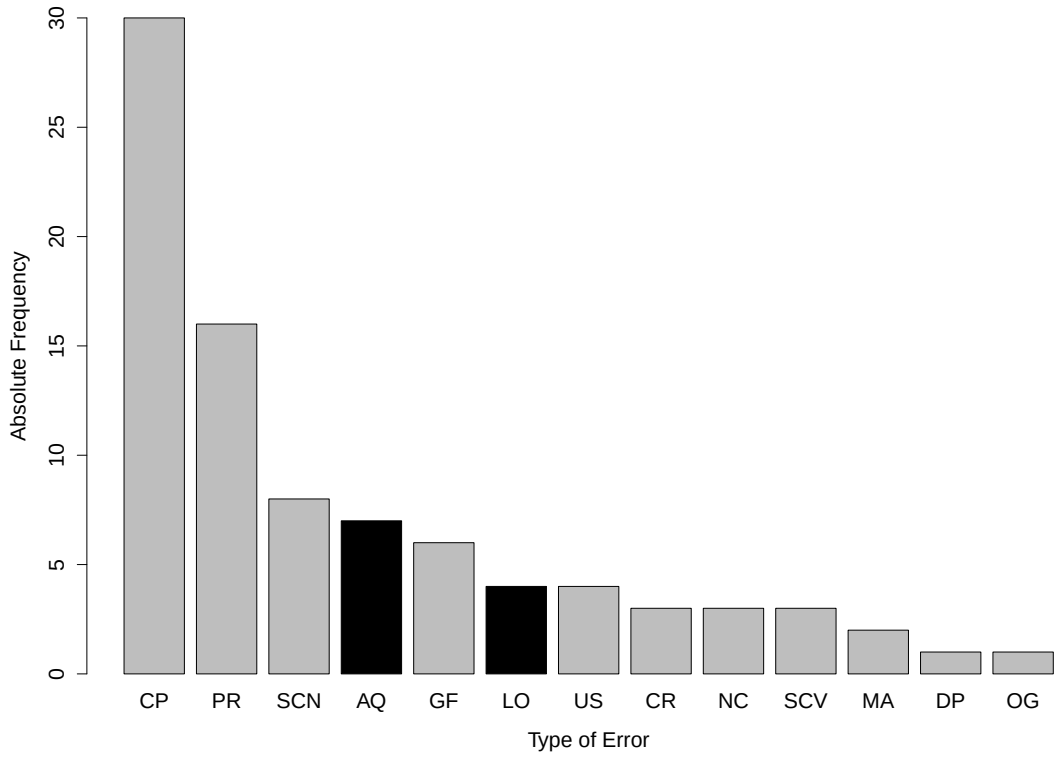

Figure 5.1.: Absolute frequencies of different types of errors.

separates the head noun and the relative clause. In this case, the simple heuristics from section 3.3 fail to introduce a complex NP and the system targets only *zu einer Zeit*, which leads to an ungrammatical question. Other parser problems seem avoidable given more training data, for example, false multi-word proper nouns (8-a) and some cases of too broad (8-c) or too narrow phrases (8-c)[5].

(8) a. Zuletzt überrundete [$_{\text{MPN}}$ Mercedes-Benz Audi].
b. Auswirkungen [$_{\text{PP}}$ aus dem VW-Abgasskandal spüre Daimler] nicht.
c. Die Toten lagen [$_{\text{PP}}$ in zwei etwa 20 Meter] [$_{\text{NP}}$ voneinander entfernten Kreisen - dort], wo die Sprengsätze explodiert waren.

The second largest bar in Figure 5.1 belongs to errors related to unresolved pro-forms and conjunctive adverbs. The pro-forms are mostly pronominal adverbs (9-a)–(9-c) and sometimes possessive pronouns (9-d). An example for a conjunctive adverb taken out of context can also be found in (9-d).

(9) a. Wer lag zur Jahresmitte darunter? – die Rivalen BMW und Audi
b. Wer sagte Polizei und Justiz erhielten zur Aufklärung schwerster Straftaten damit ein zusätzliches Instrument? – Maas

[5]In (8-c) also the NP that ends with an adverb is very atypical.

c. Wer lässt zudem Stellungen der Extremisten-Miliz Islamischer Staat (IS) in Syrien bombardieren?
d. Wonach lässt sich eine Speicherung ihrer Daten allerdings nicht verhindern? – nach Angaben der Regierung

Excluded from this category are errors caused by unresolved or falsely resolved third person personal pronouns, which were counted separately as coreference resolution errors (CR). The eight nouns with unknown or false semantic category were named entities that the NER system did not recognize (10-a), morphologically complex nouns (10-b), or a combination of both (10-c).

(10) a. Daimler, BMW
b. Telekommunikationsanbieter
c. Reuters-Reporter, FDP-Vize, FDP-Vizechef

A false semantic category of the nominal head of an answer phrase usually results in a *was* question instead of a *wer* question.

Problems with the mapping from answer phrases to question phrases affected 11 questions (cf. the black bars in the graph). These problems are due to deficiencies of the QG system itself. Some of the generalizations encoded in the answer-question mappings do not hold for all cases, for example, the nominal heads of the answer phrases in (11-a) and (11-b) belong to the semantic group *Person*, but the masculine interrogative pronoun is suboptimal in these cases. In (11-c) *die Türkei* is annotated as *Location*, but via conventional metonymy, in this context it actually denotes a political entity or a group of people (the Turkish government, the Turkish military or the soldiers carrying out the attacks). Thus, the appropriate question word would be *wer* instead of *was*.

(11) a. Von wem sprach Gesundheitsminister Mehmet Müezzinoglu? – von 86 Toten und fast 190 Verletzten
b. Wer lag in zwei etwa 20 Meter voneinander entfernten Kreisen - dort, wo die Sprengsätze explodiert waren? – die Toten
c. Was beteiligt sich gleichzeitig an den Luftangriffen gegen die radikal-islamische IS-Miliz in Syrien? – die Türkei

As expected from the discussion in section 4.3.4, the biggest problems for the linearization component are caused by adverbs (or elements that are labeled as adverbs by the tagger, such as particles), especially in combination with problematic constituency parses: The question in (12-b) is suboptimal for several reasons, but the first problem is the position of *selbst*. The dependency parser labeled *selbst* in (12-a) as postnominal modifier, however, in the constituency parse the word appears as adverb directly under the S

node and thus is not moved out of the prefield with the PP. We could try to move *selbst* out of the prefield after moving the PP, but according to Table 4.9, it would end up before the PP *für Daimler*. In the parse for (12-c), both *jetzt* and *nur* are adverbs under the S node, whereas *noch* belongs to the subsequent PP. In the question formation process, the PP in the prefield is moved behind the two adverbs, between *nur* and *noch*, changing the semantics of the question (*wegen der Dauer-Krise in Brasilien und sinkender Lkw-Verkaufszahlen in Indonesion* now is in the scope of *nur*).

(12) a. Für Daimler selbst soll dies aber keine Auswirkungen haben.
b. Was selbst soll dies aber für Daimler haben? – keine Auswirkungen
c. Wegen der Dauer-Krise in Brasilien und sinkender Lkw-Verkaufszahlen in Indonesien rechnet Daimler jetzt nur noch mit einem leichten statt einem deutlichen Absatzzuwachs in der Truck-Sparte.
d. Wer rechnet jetzt nur wegen der Dauer-Krise in Brasilien und sinkender Lkw-Verkaufszahlen in Indonesien noch mit einem leichten statt einem deutlichen Absatzzuwachs in der Truck-Sparte? – Daimler

Six low-quality questions were (at least partly) caused by false grammatical functions. In the ungrammatical question (1-b), the number of the verb does not agree with the subject (*zwei Insider*) because the (dative) answer phrase *der Nachrichtenagentur Reuters* in (13-a) was annotated as subject. Whenever the answer phrase is the subject of the sentence, the number of the verb is changed to singular – here, this breaks the subject agreement. In (13-b), the accusative object was annotated as modifier, thus the system analyzed it as durative temporal NP according to Table 4.4. The dative object *Daimler* in (13-c) was misanalyzed by the dependency parser (as subject), the RFTagger (as nominative NP) and the named-entity recognizer (as unknown entity) – the result is a *was* question instead of a *wem* question.

(13) a. Zwei Insider sagten der Nachrichtenagentur Reuters, die Behörden gingen Berichten über einen Selbstmordanschlag nach.
b. Wie lange wird die Unternehmen die Umsetzung kosten? – einen mittleren dreistelligen Millionenbetrag
c. Was spielt die hohe Nachfrage nach Autos mit dem Stern in China in die Hände? – Daimler

Four questions were vague not because of unresolved pro-forms but because of the underspecified semantics of their source sentence, which does not allow it to be taken out of context, for example:

(14) a. Was durfte nicht verwertet werden? – Daten von Berufsgeheimnisträgern

wie Anwälten oder Journalisten

b. Was ist tabu? – der E-Mail-Verkehr

The remaining errors in Figure 5.1 seem rather marginal, but some of them are connected to specific linguistic phenomena that are represented only very sparsely in the evaluation sample. For example, we have three coreference resolution errors but only five third person personal pronouns in all three texts. An example for a low-quality question caused by a failure of the coreference resolution system is the following question:

(15) Wer sagte ihre Wahlkampfauftritte ab? – Erdogan, Davutoglu und der Chef der oppositionellen Partei CHP, Kemal Kilicdaroglu

The gender of the possessive pronoun *ihre* should have been adjusted to that of the interrogative pronoun *wer*, but the system did not know that *ihre* corefers with the answer phrase. The question thus falsely implies that somebody canceled the campaign appearances of another (female) person. To be able to say more about how coreference resolution influences the performance of the QG system, I performed an additional informal test on a text with a large number of third person personal pronouns, which I report in section 5.3. The three newswire texts also contain only few embedded clauses, thus the number of overgeneration errors is low (cf. section 4.2.3), and few instances of phrases with non-compositional semantics, see for example the unacceptable question in (16).

(16) Worin sitzen die Schwaben BMW? – im Nacken

On the other hand, problems with the semantic category of the verb, as in question (17), where *stattfinden* was mistaken for the verb *finden* of the semantic type *Kognition* and consequently *der Angriff* was misanalyzed as human, are actually only a marginal problem.

(17) Wer fand zu einer Zeit statt, in der die türkischen Sicherheitskräfte gegen Kurden- und Islamisten-Gruppen vorgehen? – der Angriff

5.3. Coreference Resolution Experiment

Since the three newswire texts contained only few third person personal pronouns, but the resolution of coreferences is so important to any German QG system, I decided to additionally run the system on a simplified version of Grimm's "Hänsel und Gretel". The reason for choosing this fairy tale was that I expected to find a number of different third person personal pronouns referring to Hänsel, Gretel, both children, the parents and the

witch. I manually split coordinated clauses (to allow for more syntactic extractions) and some coordinated verb phrases (to introduce more pronouns) into separate sentences and replaced outdated language.

A first informal analysis revealed that coreference resolution often helps generating the correct question phrase but also frequently introduces errors if a pronoun needs to be resolved in the question itself. In (18), the antecedent identified by the coreference resolution tool allowed the QG system to choose the correct question word *wer* instead of *was*.

(18) a. Sie hatten mit angehört, was die Mutter zum Vater gesagt hatte.
b. Wer hatte mit angehört, was die Mutter zum Vater gesagt hatte? – die zwei Kinder

The same is true for the example in (19), but in this case, the antecedent is wrong[6]. Only the fact that Gretel and the witch both are persons prevents the system from selecting the wrong question word.

(19) a. Nun fing die Alte an in dem heißen Backofen zu schreien und zu jammern. Gretel aber lief fort. Sie musste elendiglich verbrennen.
b. Wer musste elendiglich verbrennen? – Gretel

The correct question word in (20) only was chosen because the semantic type of the verb (*Kognition*) in this case overwrites the wrong semantic type of the subject (cf. Table 4.2).

(20) a. Er dachte, es wäre doch besser, wenn du den letzten Bissen mit deinen Kindern teiltest.
b. Wer dachte, es wäre doch besser, wenn du den letzten Bissen mit deinen Kindern teiltest? – der Weg

In (21), an antecedent is picked that does not agree in number with the verb (*Hänsel* instead of *Hänsel und Gretel*), which leads to an ungrammatical question.

(21) a. Danach fanden sie bald ihre Heimat.
b. Was fanden Hänsel bald danach? – ihre Heimat

[6]This thesis is only concerned with generating good questions, but if we tried to automatically assess the answers to these questions based on the answer phrases in the text, a false antecedent would be problematic even if the correct question word was chosen.

5.4. Discussion

The evaluation shows that a large share of the questions generated by the system are already of high quality: 10% of all questions were ungrammatical, 17% unacceptable, but questions with optimal score make up the largest group with 44%, which is in a similar range as the results obtained by previous systems. Ali et al. (2010) report a precision[7] of 0.587; on average, 42% of the top 10 ranked document-level questions generated by Heilman (2011) and 46.5% of Chali and Hasan's (2015) top 15% were rated acceptable[8] in their intrinsic evaluations. However, Heilman (2011) notes that "the system performs best on Britannica Elementary articles, where it achieves 56% precision-at-10, compared to 39% for Wikipedia articles and 36% for Britannica articles" (p. 105); we saw a similar drop in performance for the text from the business section (although this might have been by chance, given the amount of text). This brings us back to the warning against comparing quantitative results across systems from section 2.2. Both the input and the evaluation schemes differ greatly, even across similar systems, and the evaluation sample used in this thesis is too small and not representative. To obtain comparable results, independent judges would have to rate questions generated from a broader, random sample according to one of the previously used rating schemes[9].

Incorrect constituency parses and unresolved pro-forms as well as certain context-dependent adverbs have been identified as the main problems. To a certain extend, the quality of the parses may be improved with more and better (domain-specific) training data, but there are limits. Another promising way of reducing parser errors and improving the system's general performance on linguistically complex data could be the addition of an independent text simplification module. Syntactic parsers usually achieve better results on shorter sentences, as the combinatorial space for syntactic structures (e.g., with different PP attachments) is much smaller. If this additional module was also capable of extracting semantically entailed and presupposed statements like the extraction algorithm in Heilman (2011), we could also extend the set of potential answer phrases and ask semantically deeper questions. To solve the second problem, we need a coreference resolution system that resolves not only personal pronouns but also pronominal adverbs; and to handle conjunctive adverbs and the word order problems related to adverbials in general, the system needs access to more fine-grained information on different subclasses of adverbials. Another weak point is the named-entity recognizer

[7] Good questions in this case need to be grammatical and related to a set of given questions.

[8] Heilman (2011) assessed acceptability "according to general linguistic factors such as grammaticality"(p. 103). For Chali and Hasan (2015), a question is acceptable if it "shows no deficiency in terms of the criteria considered for topic relevance and syntactic correctness" (p. 14) – the criteria for topic relevance include semantic correctness, the correctness of the question type and referential clarity (p. 12).

[9] The choice depends on the system with which we want to compare our system.

– although the problems with first names have already been fixed, it still performs rather poorly. Some problems could be traced back to suboptimal linguistic generalizations in the answer-question mapping and the model of the unmarked word order (although problems with the latter often occur in conjunction with deficient annotations concerning the constituency structure, grammatical roles, morphological features and adverbial subclasses). The linguistic rules concerned should be revised and tested again against new data. The number of errors related to false grammatical functions seems to suggest that the dependency parser is less reliable than the morphological tagger, however, the real reason for the relatively low number of errors caused by false morphological analyses is that the system trusts the dependency parser over the morphological tagger, thus problems related to the latter are compensated by the former.

The additional coreference experiment showed a mixed picture: The resolved pronouns helped avoiding false question phrases but also introduced errors when pronouns needed to be resolved in the question itself. The examples in (20) and (21) show that, while we cannot identify better antecedents than the coreference resolution tool, we can rule out some false antecedents using the linguistic annotation of the sentence in which the pronoun occurs. If the antecedent does not match the number of the verb or is incompatible with the semantic role required by the verb, the system should not use it to generate a question.

Questions generated for answer phrases from the prefield were of higher quality than other questions. A radical way of using this observation to boost the precision of the system could be to only generate questions for prefield constituents. However, this way we would miss out on many interesting questions (especially if prefields are frequently occupied by conjunctive adverbs like *dann*, *außerdem* or *also*).

6. Conclusions and Future Work

In this thesis, I presented the first system that dynamically generates syntactic questions for German texts. A natural language pipeline was put together from existing tools to automatically annotate the input with information from the domains of morphology, syntax, lexical semantics and discourse. Four major components have been developed to address language-specific challenges of question generation: a mapping from answer phrases to question phrases, a linearization component modeling the unmarked word order, a component that integrates antecedent information obtained from a coreference resolution system and a simple inflection model based on a lemmatizer's lexicon. There is no part of the system that could not be improved: The answer-question mapping does not cover all possible targets (e.g., phrases under finite subordinates; possessives; attributive adjectives, adverbs and participles; adjectives and adverbs in predicative or adverbial function), overgenerates for embedded clauses and has some inaccuracies, as the evaluation has shown; the linearization is completely static and currently does not have enough information to appropriately handle different kinds of adverbials; antecedents have a high error rate and should be filtered based on agreement information; the inflection model does not cover the inflection classes of German adjectives. Still, the evaluation results were promising: We saw a large variety of question phrases and almost half of all generated questions received the highest quality score.

Most problems in lower-scored questions were due to parser errors and context-dependent expressions that should have been resolved or deleted. Only relatively few problematic questions resulted from errors of the core of the system itself. Parser-related errors might be reduced with an independent text simplification module. To avoid overgenerating questions for embedded clauses, the system needs access to more linguistic features, such as the argument frames of verbs or the exact grammatical functions of embedded clauses and their relation to correlates in the sentence. In order to further improve the overall precision of the system, a ranking model could be used to filter out ill-formed output. This has already proven to be a successful strategy: Heilman and Smith (2010) report that statistical ranking "approximately doubled the acceptability of the top-ranked questions" (p. 616), Heilman (2011) is able to replicate this result[1].

[1]Chali and Hasan (2012, 2015) report similar results for their ranking based on topic relevance and syntactic similarity scores. Mannem et al. (2010) rank according to linguistic features indicative of question quality, Yao, Bouma, and Zhang (2012) combine two statistical models, but both did not

To generate more specific questions and to introduce some lexical variation, it would be interesting to explore certain lexical semantic relations in GermaNet. For example, instead of question (1-a), the system could generate the more specific question (1-b) using the pattern '*welche* + hypernym' to form the question phrase.

(1) Asya spricht Persisch.

a. Was spricht Asya?

b. Welche Sprache spricht Asya?

For avoiding errors related to incomplete linguistic generalizations, crowdsourcing experiments might be an interesting option. We could use the NLP pipeline from this thesis to identify potential answer phrases and ask workers to annotate appropriate substitutive question phrases. Based on these data, an optimal mapping between linguistic features of answer phrases and question phrases could be found via machine learning.

Finally, it would, of course, be interesting to approach question generation for German from a completely different angle – by adopting either Yao and Zhang's (2010) semantic tranformations or Curto et al.'s (2012) lexico-syntactic patterns.

evaluate the ranker's influence on their system's performance.

References

Agarwal, M., Shah, R., & Mannem, P. (2011). Automatic question generation using discourse cues. In *Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications* (pp. 1–9).

Albert, S., Anderssen, J., Bader, R., Becker, S., Bracht, T., Brants, S., ... Uszkoreit, H. (2003). *TIGER Annotationsschema.* Universität des Saarlandes, Universität Stuttgart and Universität Potsdam. Retrieved August 28, 2015, from `http://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/TIGERCorpus/annotation/tiger_scheme-syntax.pdf`

Aldabe, I., de Lacalle, M. L., Maritxalar, M., Martinez, E., & Uria, L. (2006). ArikIturri: an automatic question generator based on corpora and NLP techniques. In *Intelligent tutoring systems* (pp. 584–594).

Aldabe, I., Gonzalez-Dios, I., Lopez-Gazpio, I., Madrazo, I., & Maritxalar, M. (2013). Two approaches to generate questions in basque. *Procesamiento del lenguaje natural*, *51*, 101–108.

Aldabe, I., Maritxalar, M., & Soraluze, A. (2011). Question generation based on numerical entities in basque. In *AAAI Fall Symposium: Question Generation.*

Ali, H., Chali, Y., & Hasan, S. A. (2010). Automation of question generation from sentences. In *Proceedings of the 3rd Workshop on Question Generation* (pp. 58–67).

Andersson, S.-G., & Kvam, S. (1984). Satzverschränkung im heutigen Deutsch. *Narr, Tübingen.*

Baumann, S., & Riester, A. (2012). Referential and lexical givenness: Semantic, prosodic and cognitive aspects. *Prosody and meaning*, *25*, 119–161.

Bernhard, D., De Viron, L., Moriceau, V., & Tannier, X. (2012). Question generation for French: collating parsers and paraphrasing questions. *Dialogue and Discourse*, *3*(2), 43–74.

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with Python.* O'Reilly Media, Inc.

Bloom, B. S. (Ed.). (1956). *Taxonomy of educational objectives: The classification of educational goals. Handbook I: Cognitive domain.* Longman.

Bohnet, B. (2010). Very high accuracy and fast dependency parsing is not a contradiction. In *Proceedings of the 23rd International Conference on Computational Linguistics*

(pp. 89–97).

Brown, J. C., Frishkoff, G. A., & Eskenazi, M. (2005). Automatic question generation for vocabulary assessment. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing* (pp. 819–826).

Büring, D. (1994). Mittelfeldreport V. In *Was determiniert Wortstellungsvariation?* (pp. 79–96). Springer.

Chali, Y., & Hasan, S. A. (2012). Towards automatic topical question generation. In *Proceedings of COLING 2012: Technical papers* (pp. 475–492).

Chali, Y., & Hasan, S. A. (2015). Towards topic-to-question generation. *Computational Linguistics*, *41*(1), 1–20.

Chen, W., Aist, G., & Mostow, J. (2009). Generating questions automatically from informational text. In S. D. Craig & D. Dicheva (Eds.), *Proceedings of the 2nd Workshop on Question Generation (AIED 2009)* (pp. 17–24).

Ciaramita, M., & Altun, Y. (2006). Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (pp. 594–602).

Collins, M. J. (1999). *Head-driven statistical models for natural language parsing* (Unpublished doctoral dissertation). University of Pennsylvania, Philadelphia, PA, USA.

Curto, S., Mendes, A., & Coheur, L. (2012). Question generation based on lexico-syntactic patterns learned from the web. *Dialogue & Discourse*, *3*(2), 147–175.

Das, R., & Elikkottil, A. (2010). Auto-summarizer to aid a q/a system. *International Journal of Computer Applications*, *1*(19), 113–117.

Faruqui, M., & Padó, S. (2010). Training and evaluating a German named entity recognizer with semantic generalization. In *Proceedings of KONVENS 2010.* Saarbrücken, Germany.

Frey, W., & Pittner, K. (1998). Zur Positionierung der Adverbiale im deutschen Mittelfeld. *Linguistische Berichte*(176), 489–534.

Frey, W., & Pittner, K. (1999). Adverbialpositionen im deutsch-englischen Vergleich. In M. Doherty (Ed.), *Sprachspezifische Aspekte der Informationsverteilung* (pp. 14–40).

Gallmann, P. (2015). *Syntax-Theorie: w-Bewegung.* FSU Jena. Retrieved August 15, 2015, from `http://www2.uni-jena.de/philosophie/germsprach/syntax/2//doc/skript/WissBlock_G.pdf` (lecture notes)

Garg, P., & Bedi, E. C. S. (2013). Automatic question generation system from Punjabi text using hybrid approach. *International Journal of Computer Trends and Technology (IJCTT)*, *21*(3), 130–133.

Garg, S., & Goyal, V. (2013). System for generating questions automatically from given

Punjabi text. *International journal of computer Science and mobile computing*, 324–327.

Gates, D. M. (2008). *Automatically generating reading comprehension look-back strategy questions from expository texts* (Unpublished master's thesis). Carnegie Mellon University, The address of the publisher.

Graesser, A. C., Langston, M. C., & Lang, K. L. (1992, April). Designing educational software around questioning. *Journal of Artificial Intelligence in Education*, *3*(2), 235–241.

Graesser, A. C., & Person, N. K. (1994). Question asking during tutoring. *American Educational Research Journal*, *31*(1), 104–137.

Graesser, A. C., Rus, V., D'Mello, S. K., Jackson, G. T., Robinson, D. H., & Schraw, G. (2008). AutoTutor: Learning through natural language dialogue that adapts to the cognitive and affective states of the learner. *Recent innovations in educational technology that facilitate student learning*, 95–125.

Gütl, C., Lankmayr, K., Weinhofer, J., & Höfler, M. (2011). Enhanced automatic question creator – EAQC: Concept, development and evaluation of an automatic test item creation tool to foster modern e-education. *Electronic Journal of e-Learning*, *9*(1), 23–38.

Hamp, B., & Feldweg, H. (1997). GermaNet - a lexical-semantic net for German. In *Proceedings of ACL workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications* (pp. 9–15).

Harabagiu, S., Hickl, A., Lehmann, J., & Moldovan, D. (2005). Experiments with interactive question-answering. In *Proceedings of the 43rd annual meeting on Association for Computational Linguistics* (pp. 205–214).

Heilman, M. (2011). *Automatic factual question generation from text* (Unpublished doctoral dissertation). Carnegie Mellon University.

Heilman, M., & Smith, N. A. (2009). *Question generation via overgenerating transformations and ranking* (Tech. Rep.). Pittsburgh, PA: Language Technologies Institute, School of Computer Science, Carnegie Mellon University.

Heilman, M., & Smith, N. A. (2010). Good question! Statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 609–617).

Helbig, G., & Buscha, J. (1987). *Deutsche Grammatik* (10th ed.). Verlag Enzyklopädie Leipzig.

Henrich, V., & Hinrichs, E. W. (2010). GernEdiT - the GermaNet editing tool. In *Proceedings of the ACL 2010 System Demonstrations* (pp. 19–24).

Henrich, V., & Hinrichs, E. W. (2012). A comparative evaluation of word sense

disambiguation algorithms for German. In *LREC* (pp. 576–583).

Hoberg, U. (1977). Die Wortstellung in der geschriebenen deutschen Gegenwartssprache: Untersuchungen zur Elementfolge im einfachen Satz. *Mitteilungen des Instituts für deutsche Sprache.*

Hofmann, U. (1994). *Zur Topologie im Mittelfeld: pronominale und nominale Satzglieder* (Vol. 307). Walter de Gruyter.

Höhle, T. (1982). Explikation für "normale betonung" und "normale wortstellung". In *Satzglieder im deutschen.* Gunther Narr Verlag.

Jacobs, J. (1993). Integration. In M. Reis (Ed.), *Wortstellung und informationsstruktur* (Vol. 306, pp. 63–116). Walter de Gruyter.

Jacobs, J. (1999). Informational autonomy. *Focus. Linguistic, cognitive and computational perspectives*, 56–81.

Jouault, C., & Seta, K. (2013). Building a semantic open learning space with adaptive question generation support. In *Proceedings of the 21st International Conference on Computers in Education* (pp. 41–50).

Jouault, C., & Seta, K. (2014). Content-dependent question generation for history learning in semantic open learning space. In S. Trausan-Matu, K. Boyer, M. Crosby, & K. Panourgia (Eds.), *Intelligent tutoring systems* (Vol. 8474, p. 300-305). Springer International Publishing.

Jouault, C., Seta, K., & Hayashi, Y. (2015). Quality of lod based semantically generated questions. In C. Conati, N. Heffernan, A. Mitrovic, & M. F. Verdejo (Eds.), *Artificial intelligence in education* (Vol. 9112, p. 662-665). Springer International Publishing.

Kalady, S., Elikkottil, A., & Das, R. (2010). Natural language question generation using syntax and keywords. In *Proceedings of the 3rd Workshop on Question Generation* (pp. 1–10).

Kaur, J., & Bathla, A. K. (2015). Automatic question generation from hindi text using hybrid approach. *International Journal of Advanced Technology in Engineering and Science*, *3*(1), 435–442.

Klein, D., & Manning, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics* (Vol. 1, pp. 423–430).

Klein, N. M., Gegg-Harrison, W. M., Sussman, R. S., Carlson, G. N., & Tanenhaus, M. K. (2009). Weak definite noun phrases: rich, but not strong, special, but not unique. In U. Sauerland & K. Yatsihshiro (Eds.), *Semantics and pragmatics, from experiment to theory.* Palgrave Macmillan.

Klenner, M., & Tuggener, D. (2011). An incremental entity-mention model for coreference resolution with restrictive antecedent accessibility. In *Recent Advances in Natural*

Language Processing (RANLP 2011) (pp. 178–185).

Koskeniemmi, K., & Haapalainen, M. (1994). Gertwol–lingsoft oy. *Linguistische Verifikation: Dokumentation zur Ersten Morpholympics*, 121–140.

Krifka, M. (2008). Basic notions of information structure. *Acta Linguistica Hungarica*, *55*(3-4), 243–276.

Kunichika, H., Katayama, T., Hirashima, T., & Takeuchi, A. (2001). Automated question generation methods for intelligent English learning systems and its evaluation. In *Proceedings of the International Conference on Computers in Education - ICCE* (pp. 1117–1124).

Lane, H. C., & VanLehn, K. (2005). Teaching the tacit knowledge of programming to novices with natural language tutoring. *Computer Science Education*, *15*(3), 183–201.

Le, N.-T., Kojiri, T., & Pinkwart, N. (2014). Automatic question generation for educational applications – the state of art. In T. van Do, H. A. L. Thi, & N. T. Nguyen (Eds.), *Advanced computational methods for knowledge engineering* (Vol. 282, p. 325-338). Springer International Publishing.

Le, N.-T., & Pinkwart, N. (2015). Evaluation of a question generation approach using semantic web for supporting argumentation. *Research and Practice in Technology Enhanced Learning*, *10*(1).

Lenerz, J. (1977). *Zur Abfolge nominaler Satzglieder im Deutschen* (Vol. 5). TBL-Verlag Narr.

Levy, R., & Andrew, G. (2006). Tregex and Tsurgeon: tools for querying and manipulating tree data structures. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006)* (pp. 2231–2234).

Liu, M., Calvo, R. A., & Rus, V. (2012). G-asks: An intelligent automatic question generation system for academic writing support. *Dialogue and Discourse: Special Issue on Question Generation*, *3*(2), 101–124.

Liu, M., Calvo, R. A., & Rus, V. (2014). Automatic generation and ranking of questions for critical review. *Journal of Educational Technology & Society*, *17*(2), 333–346.

Lötscher, A. (1984). Satzgliedstellung und funktionale Satzperspektive. *Pragmatik in der Grammatik*, 118–151.

Lühr, R. (1988). Zur Satzverschränkung im heutigen Deutsch. *Groninger Arbeiten zur Germanistischen Linguistik*, *29*, 74–87.

Mannem, P., Prasad, R., & Joshi, A. (2010). Question generation from paragraphs at upenn: Qgstec system description. In *Proceedings of the 3rd Workshop on Question Generation* (pp. 84–91).

Mitkov, R., An Ha, L., & Karamanis, N. (2006). A computer-aided environment for generating multiple-choice test items. *Natural Language Engineering*, *12*(02),

177–194.

Mostow, J., & Chen, W. (2009). Generating instruction automatically for the reading strategy of self-questioning. In *Proceeding of the Conference on Artificial Intelligence in Education* (pp. 465–472).

Mostow, J., Tobin, B., & Cuneo, A. (2002). Automated comprehension assessment in a reading tutor. In *Proceedings of the ITS 2002 Workshop on Creating Valid Diagnostic Assessments* (pp. 52–63).

Müller, G. (1999). Optimality, markedness, and word order in German. *Linguistics*, *37*(5), 777–818.

Nielsen, R. D., Buckingham, J., Knoll, G., Marsh, B., & Palen, L. (2008). A taxonomy of questions for question generation. In *Proceedings of the workshop on the question generation shared task and evaluation challenge.*

Olney, A. M., Graesser, A. C., & Person, N. K. (2012). Question generation from concept maps. *Dialogue and Discourse*, *3*(2), 75–99.

Pal, S., Mondal, T., Pakray, P., Das, D., & Bandyopadhyay, S. (2010). Qgstec system description – jugqq: A rule based approach. In *Proceedings of the 3rd Workshop on Question Generation* (pp. 76–79).

Pittner, K., & Berman, J. (2007). *Deutsche Syntax* (2nd ed.).

Piwek, P., & Boyer, K. E. (2012). Varieties of question generation: Introduction to this special issue. *Dialogue and Discourse*, *3*, 1–9.

Piwek, P., & Stoyanchev, S. (2010). Question generation in the coda project. In *Proceedings of the 3rd Workshop on Question Generation* (pp. 29–34).

Polenz, P. (2008). *Deutsche Satzsemantik: Grundbegriffe des Zwischen-den-Zeilen-Lesens.* Walter de Gruyter.

Prince, E. F. (1981). Toward a taxonomy of given-new information. *Radical pragmatics*, 223–255.

Rafferty, A. N., & Manning, C. D. (2008). Parsing three German treebanks: Lexicalized and unlexicalized baselines. In *Proceedings of the Workshop on Parsing German* (pp. 40–46).

Reis, M. (1987). Die Stellung der Verbargumente im Deutschen: Stilübungen zum Grammatik: Pragmatik-Verhältnis. In *Sprache und Pragmatik* (Vol. 5). Almqvist & Wiksell International.

Rösiger, I., & Riester, A. (2015). Using prosodic annotations to improve coreference resolution of spoken text. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers)* (pp. 83–88).

Rus, V., Cai, Z., & Graesser, A. C. (2008). Question generation: Example of a multi-year evaluation campaign. In *Proceedings*

of the 1st Workshop on Question Generation. Retrieved September 20, 2015, from http://www.researchgate.net/publication/228948043_Question_Generation_Example_of_A_Multi-year_Evaluation_Campaign

Rus, V., Wyse, B., Piwek, P., Lintean, M., Stoyanchev, S., & Moldovan, C. (2010). Overview of the First Question Generation Shared Task Evaluation Challange. In *Proceedings of the 3rd Workshop on Question Generation* (pp. 45–57).

Rus, V., Wyse, B., Piwek, P., Lintean, M., Stoyanchev, S., & Moldovan, C. (2012). A detailed account of the first question generation shared task evaluation challenge. *Dialogue and Discourse*, *3*(2), 177–204.

Schiller, A., Teufel, S., Stöckert, C., & Thielen, C. (1999). *Guidelines für das Tagging deutscher Textcorpora mit STTS.* Universitäten Stuttgart und Tübingen. Retrieved August 27, 2015, from http://www.ims.uni-stuttgart.de/forschung/ressourcen/lexika/TagSets/stts-1999.pdf

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing* (Vol. 12, pp. 44–49).

Schmid, H. (1995). Improvements in part-of-speech tagging with an application to German. In *Proceedings of the ACL SIGDAT-Workshop.*

Schmid, H., Fitschen, A., & Heid, U. (2004). SMOR: A German computational morphology covering derivation, composition and inflection. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC 2004).*

Schmid, H., & Laws, F. (2008). Estimation of conditional probabilities with decision trees and an application to fine-grained POS tagging. In *Proceedings of the 22nd International Conference on Computational Linguistics* (Vol. 1, pp. 777–784).

Seeker, W., & Kuhn, J. (2012). Making ellipses explicit in dependency conversion for a German treebank. In *Lrec* (pp. 3132–3139).

Sennrich, R., & Kunz, B. (2014). Zmorge: A German morphological lexicon extracted from wiktionary. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014).*

Sennrich, R., Volk, M., & Schneider, G. (2013). Exploiting synergies between open resources for German dependency parsing, pos-tagging, and morphological analysis. In *Ranlp* (pp. 601–609).

Singh, R., Gulwani, S., & Rajamani, S. K. (2012). Automatically generating algebra problems. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence* (pp. 1620–1627).

Singhal, R., Henz, M., & Goyal, S. (2015). A framework for automated generation of questions based on first-order logic. In C. Conati, N. Heffernan, A. Mitrovic, &

M. F. Verdejo (Eds.), *Artificial intelligence in education* (Vol. 9112, p. 776-780). Springer International Publishing.

Sternefeld, W. (2007). *Eine morphologisch motivierte generative Beschreibung des Deutschen* (2nd ed., Vol. 1). Stauffenburg.

Susanti, Y., Iida, R., & Tokunaga, T. (2015). Automatic generation of english vocabulary tests. In *Proceedings of the 7th International Conference on Computer Supported Education (CSEDU 2015)* (pp. 77–78).

Toutanova, K., & Manning, C. D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *Proceedings of the 2000 Joint SIGDAT Conference on Empirical methods in Natural Language Processing and Very Large Corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics* (Vol. 13, pp. 63–70).

Varga, A., & Ha, L. A. (2010). Wlv: A question generation system for the qgstec 2010 task b. In *Proceedings of the 3rd Workshop on Question Generation* (pp. 80–83).

Wyse, B., & Piwek, P. (2009). Generating questions from openlearn study units. In *Proceedings of the 2nd Workshop on Question Generation, AIED 2009* (pp. 66–73).

Yao, X., Bouma, G., & Zhang, Y. (2012). Semantics-based question generation and implementation. *Dialogue & Discourse*, *3*(2), 11–42.

Yao, X., Tosch, E., Chen, G., Nouri, E., Artstein, R., Leuski, A., ... Traum, D. (2012). Creating conversational characters using question generation tools. *Dialogue and Discourse*, *3*(2), 125–146.

Yao, X., & Zhang, Y. (2010). Question generation with minimal recursion semantics. In *Proceedings of the 3rd Workshop on Question Generation* (pp. 68–75).

A. Tags and Labels

A.1. Small STTS (1999)

A.1.1. Original Set

Tag	Description	Examples
ADJA	attributive adjective	[das] große [Haus]
ADJD	adverbial or predicative adjective	[er fährt] schnell [er ist] schnell
ADV	adverb	schon, bald, doch
APPR	preposition; left circumposition	in [der Stadt], ohne [mich]
APPRART	preposition with article	im [Haus], zur [Sache]
APPO	postposition	[ihm] zufolge, [der Sache] wegen
APZR	right circumposition	[von jetzt] an
ART	definite or indefinite article	der, die, das ein, eine, ...
CARD	cardinal number	zwei [Männer], [im Jahre] 1994
FM	foreign material	[Er hat das mit “] A big fish [” übersetzt]
ITJ	interjection	mhm, ach, tja
KOUI	subordinating conjunction with *zu* and infinitive	um [zu leben], anstatt [zu fragen]
KOUS	subordinating conjunction with sentence	weil, dass, damit, wenn, ob
KON	coordinating conjunction	und, oder, aber
KOKOM	comparative conjunction	als, wie
NN	regular noun	Tisch, Herr, [das] Reisen

NE	proper noun	Hans, Hamburg, HSV
PDS	substituting demonstrative pronoun	dieser, jener
PDAT	attributive demonstrative pronoun	jener [Mensch]
PIS	substituting indefinite pronoun	keiner, viele, man, niemand
PIAT	attributive demonstrative pronoun	jener [Mensch]
PIDAT	attributive indefinite pronoun without determiner	kein [Mensch], irgendein [Glas]
PPER	non-reflexive personal pronoun	ich, er, ihm, mich, dir
PPOSS	substituting possessive pronoun	meins, deiner
PPOSAT	attributive possessive pronoun	mein [Buch], deine [Mutter]
PRELS	substituting relative pronoun	[der Hund,] der
PRELAT	attributive relative pronoun	[der Mann,] dessen [Hund]
PRF	reflexive personal pronoun	sich, dich, mir
PWS	substituting interrogative pronoun	wer, was
PWAT	attributive interrogative pronoun	welche [Farbe], wessen [Hut]
PWAV	adverbial interrogative or relative pronoun	warum, wo, wann, worüber, wobei
PAV	pronominal adverb	dafür, dabei, deswegen, trotzdem
PTKZU	*zu* before infintive	zu [gehen]
PTKNEG	negation particle	nicht
PTKVZ	separated verb particle	[er kommt] an, [er fährt] rad
PTKANT	answer particle	ja, nein, danke, bitte
PTKA	particle with adjective or adverb	am [schönsten], zu [schnell]
SGML	SGML markup	<turnid=n002k_TS2004>
SPELL	letter sequence	S-C-H-W-E-I-K-L
TRUNC	first part of a truncated word	An- [und Abreise]
VVFIN	finite verb, full	[du] gehst, [wir] kommen [an]
VVIMP	imperative, auxiliary	komm [!]
VVINF	infinite, full	gehen, ankommen
VVIZU	infinitive with *zu*, full	anzukommen, loszulassen
VVPP	past participle, full	gegangen, angekommen
VAFIN	finite verb, auxiliary	[du] bist, [wir] werden
VAIMP	imperative, auxiliary	sei [ruhig !]
VAINF	infinitive, auxiliary	werden, sein
VAPP	perfect participle, auxiliary	gewesen
VMFIN	finite verb, modal	dürfen

VMINF	infinitive, modal	wollen
VMPP	perfect participle, modal	gekonnt, [er hat gehen] können
XY	non-word containing non-letter	3:7, H2O, D2XW3
$,	comma	,
$.	sentence-final punctuation mark	. ? ! ; :
$(	other sentence-internal punctuation mark	- [,] ( )

Table A.1.: Small STTS (Schiller et al., 1999) with additional tags SGML and SPELL (Albert et al., 2003, Appendix B).

A.1.2. TIGER Modifications

- **PIDAT** and **PIAT** are not distinguished in the TIGER annotation. For both PIAT is used.
- Prepositions are tagged as **ADV** if they modify numerals.
- **PROAV** is used instead of **PAV** – with the same meaning.

A.2. Large STTS (1999)

Attribute	Possible value	Used with
Genus	Masc, Fem, Neut	NN, NE, ADJA, ART, PPER, PPOS., PD., PI., PRELS, PWAT, PWS, APPRART
Kasus	Nom, Gen, Dat, Akk	NN, NE, ADJA, ART, PPER, PRF, PPOS., PD., PI., PRELS, PWAT
Numerus	Sg, Pl	NN, NE, ADJA, V.FIN, V.IMP, ART, PPER, PRF, PPOS., PD., PI., PRELS, PWAT, PWS
Flexion	St, Sw, Mix	NN, ADJA
Grad	Pos, Comp, Sup	ADJA, ADJD
Person	1, 2, 3	V.FIN, PPER, PRF
Tempus	Pres, Past	V.FIN
Modus	Ind, Konj	V.FIN
Definitheit	Def, Indef	ART

Table A.2.: Linguistic attributes, their values and the STTS-tags they apply to (Schiller et al., 1999, p. 8).

A.3. Node Labels

Label	Description
AA	superlative phrase with *am*
AP	adjective phrase
AVP	adverbial phrase
CAC	coordinated adpositions
CAP	coordinated adjective phrase
CAVP	coordinated adverbial phrase
CCP	coordinated complementizer
CH	chunk
CNP	coordinated noun phrase
CO	coordination
CPP	coordinated adpositional phrase
CS	coordinated sentence

CVP	coordinated verb phrase (non-finite)
CVZ	coordinated *zu*-marked infinitive
DL	discourse level constituent
ISU	idiosyncratic unit
MPN	multi-word proper noun
MTA	multi-token adjective
NM	multi-token number
NP	noun phrase
PP	adpositional phrase
QL	quasi-language
S	sentence
VP	verb phrase (non-finite)
VZ	*zu*-marked infinitive

Table A.3.: Constituent labels in the NEGRA and TIGER treebank, compiled from `http://www.coli.uni-saarland.de/projects/sfb378/negra-corpus/knoten.html` and Albert et al. (2003).

A.4. Edge Labels

Label	**Description**	**NEGRA**	**TIGER**
AC	adpositional case marker	✓	✓
ADC	adjective component	✓	✓
AG	genitive attribute	✗	✓
AMS	measure argument of an adjective/adverb	✓	✓
APP	apposition	✓	✓
AVC	adverbial phrase component	✓	✓
CC	comparative complement	✓	✓
CD	coordinating conjunction	✓	✓
CJ	conjunct	✓	✓
CM	comparative conjunction	✓	✓
CP	complementizer	✓	✓
CVC	collocational verb construction	✗	✓
DA	dative object/'free dative'	✓	✓
DH	discourse-level head	✓	✓
DM	discourse marker	✓	✓
EP	expletive *es*	✗	✓
GL	prenominal genitive	✓	✗

GR	postnominal genitive	✓	✗
HD	head	✓	✓
JU	junctor	✓	✓
MNR	postnominal modifier	✓	✓
MO	modifier	✓	✓
NG	negation	✓	✓
NK	noun kernel modifier	✓	✓
NMC	numerical component	✓	✓
OA	accusative object	✓	✓
OA2	second accusative object	✓	✓
OC	clausal object	✓	✓
OG	genitive object	✓	✓
PD	predicative	✓	✓
PG	pseudo-genitive	✓	✓
PH	place holder	✓	✓
PM	morphological particle	✓	✓
PNC	proper noun component	✓	✓
RC	relative clause	✓	✓
RE	repeated element	✓	✓
RS	reported speech	✓	✓
SB	subject	✓	✓
SBP	passivised subject (PP)	✓	✓
SP	subject or predicate	✓	✓
SVP	separable verb prefix	✓	✓
UC	(idiosyncatic) unit component	✓	✓
VO	vocative	✓	✓

Table A.4.: Edge labels according to the annotation schemes of NEGRA and TIGER, compiled from `http://www.coli.uni-saarland.de/projects/sfb378/negra-corpus/kanten.html` and Albert et al. (2003).

B. Rules

B.1. Heilman's (2011) Tregex Movement Constraints

1. `VP < (S=unmv $,, /,/)`
2. `S < PP|ADJP|ADVP|S|SBAR=unmv > ROOT`
3. `/\\.*/ < CC << NP|ADJP|VP|ADVP|PP=unmv`
4. `SBAR < (IN|DT < /[^that]/) << NP|PP=unmv`
5. `SBAR < /^WH.*P$/ << NP|ADJP|VP|ADVP|PP=unmv`
6. `SBAR <, IN|DT < (S < (NP=unmv !$,, VP))`
7. `S < (VP <+(VP) (VB|VBD|VBN|VBZ < be|being|been|is|are|was|were|am) <+(VP) (S << NP|ADJP|VP|ADVP|PP=unmv))`
8. `NP << (PP=unmv !< (IN < of|about))`
9. `PP << PP=unmv`
10. `NP $ VP << PP=unmv`
11. `SBAR=unmv [ !> VP | $-- /,/ | < RB ]`
12. `SBAR=unmv !< WHNP < (/^[^S].*/ !<< that|whether|how)`
13. `NP=unmv < EX`
14. `/^S/ < " << NP|ADJP|VP|ADVP|PP=unmv`
15. `PP=unmv !< NP`
16. `NP=unmv $ @NP`
17. `NP|PP|ADJP|ADVP << NP|ADJP|VP|ADVP|PP=unmv`
18. `@UNMV << NP|ADJP|VP|ADVP|PP=unmv`

B.2. Answer-Question Phrase Mapping for PPs

This appendix section lists the complete answer-question phrase mapping for PPs. Lines that start with the hash sign (#) are comments or commented-out lines (with ignored prepositions).

```
# prepositions governing Acc or Dat; different meanings
an-Acc-Person-*-* (QP (APPR an) (PWS wen))
```

```
an-Acc-*-OP-* (QP (PWAV woran))
an-Acc-*-MO-* (QP (PWAV wohin))
an-Dat-Person-*-* (QP (APPR an) (PWS wem))
an-Dat-*-OP-* (QP (PWAV woran))
an-Dat-*-MO-* (QP (PWAV wo))
an-Dat-Time-MO-* (QP (PWAV wann))
am-Dat-Person-*-* (QP (APPR an) (PWS wem))
am-Dat-*-OP-* (QP (PWAV woran))
am-Dat-*-MO-* (QP (PWAV wo))
am-Dat-Time-MO-* (QP (PWAV wann))

auf-Acc-Person-*-* (QP (APPR auf) (PWS wen))
auf-Acc-*-OP-* (QP (PWAV worauf))
auf-Acc-*-MO-* (QP (PWAV wohin))
auf-Dat-Person-OP-* (QP (APPR auf) (PWS wem))
auf-Dat-*-OP-* (QP (PWAV worauf))
auf-Dat-*-MO-* (QP (PWAV wo))

hinter-Acc-Person-OP-* (QP (APPR hinter) (PWS wen))
hinter-Acc-*-OP-* (QP (PWAV wohinter))
hinter-Acc-*-MO-* (QP (PWAV wohin))
hinter-Dat-Person-*-* (QP (APPR hinter) (PWS wem))
hinter-Dat-*-OP-* (QP (PWAV wohinter))
hinter-Dat-*-MO-* (QP (PWAV wo))

in-Acc-Person-*-* (QP (APPR in) (PWS wen))
in-Acc-Time-*-* (QP (PWAV wann))
in-Acc-*-*-* (QP (PWAV wohin))
in-Dat-Location-MO-* (QP (PWAV wo))
in-Dat-Person-*-* (QP (APPR in) (PWS wem))
in-Dat-Time-*-* (QP (PWAV wann))
in-Dat-*-MO-* (QP (PWAV worin))
im-Dat-Time-*-* (QP (PWAV wann))
im-Dat-Location-*-* (QP (PWAV wo))
im-Dat-*-*-* (QP (PWAV worin))

neben-Acc-Person-*-* (QP (APPR neben) (PWS wen))
neben-Acc-*-*-* (QP (APPR neben) (PWS was))
```

```
neben-Dat-Person-*-* (QP (APPR neben) (PWS wem))
neben-Dat-*-*-* (QP (APPR neben) (PWS was))

über-Acc-Person-*-* (QP (APPR über) (PWS wen))
über-Acc-*-OP-* (QP (PWAV worüber))
über-Acc-*-*-* (QP (PWAV worüber))
über-Dat-Person-*-* (QP (APPR über) (PWS wem))
über-Dat-*-OP-* (QP (PWAV worüber))
über-Dat-*-MO-* (QP (PWAV wo))

unter-Acc-Person-*-* (QP (APPR unter) (PWS wen))
unter-Dat-Person-*-* (QP (APPR unter) (PWS wem))
unter-Acc-*-*-* (QP (PWAV worunter))
unter-Dat-*-*-* (QP (PWAV wo))

vor-Acc-Person-*-* (QP (APPR vor) (PWS wen))
vor-Dat-Person-*-* (QP (APPR vor) (PWS wem))
vor-Dat-Time-*-* (QP (PWAV wann))
vor-Acc-*-*-* (QP (PWAV wovor))
vor-Dat-*-OP-* (QP (PWAV wovor))
vor-Dat-*-MO-* (QP (PWAV wo))

zwischen-Acc-*-*-* (QP (APPR zwischen) (PWS was))
zwischen-Acc-Person-*-* (QP (APPR zwischen) (PWS wen))
zwischen-Dat-*-*-* (QP (APPR zwischen) (PWS was))
zwischen-Dat-Person-*-* (QP (APPR zwischen) (PWS wem))


# genitive prepositions
abzüglich-Gen-*-*-* (QP (APPR abzüglich) (PWS wessen))
angesichts-Gen-*-*-* (QP (APPR angesichts) (PWS wessen))
anstatt-Gen-*-*-* (QP (APPR anstatt) (PWS wessen))
statt-Gen-*-*-* (QP (APPR statt) (PWS wessen))
#außerhalb-Gen-*-*-* (QP (APPR außerhalb) (PWS wessen))
außerhalb-Gen-*-*-* (QP (PWAV wo))
#bar
#behufs
bezüglich-Gen-*-*-* (QP (APPR bezüglich) (PWS wessen))
```

```
diesseits-Gen-*-*-* (QP (PWAV wo))
einschließlich-Gen-*-*-* (QP (APPR einschließlich) (PWS wessen))
entlang-Gen-*-*-* (QP (PWAV wo) (APPO entlang))
infolge-Gen-*-*-* (QP (APPR infolge) (PWS wessen))
innerhalb-Gen-*-*-* (QP (APPR innerhalb) (PWS wessen))
inmitten-Gen-*-*-* (QP (APPR inmitten) (PWS wessen))
jenseits-Gen-*-*-* (QP (APPR jenseits) (PWS wessen))
kraft-Gen-*-*-* (QP (APPR kraft) (PWS wessen))
längs-Gen-*-*-* (QP (PWAV wo) (APPO entlang))
mittels-Gen-*-*-* (QP (APPR mittels) (PWS wessen))
#ob
oberhalb-Gen-*-*-* (QP (APPR oberhalb) (PWS wessen))
seitens-Gen-*-*-* (QP (APPR seitens) (PWS wessen))
trotz-Gen-*-*-* (QP (APPR trotz) (PWS wessen))
#unbeschadet-Gen-*-*-* (QP (APPR unbeschadet) (PWS wessen))
ungeachtet-Gen-*-*-* (QP (APPR ungeachtet) (PWS wessen))
unterhalb-Gen-*-*-* (QP (APPR unterhalb) (PWS wessen))
unweit-Gen-*-*-* (QP (APPR unweit) (PWS wessen))
während-Gen-*-*-* (QP (PWAV wann))
wegen-Gen-*-*-* (QP (PWAV weswegen))
zugunsten-Gen-*-*-* (QP (APPR zugunsten) (PWS wessen))

# dative prepositions
aus-Dat-Location-*-* (QP (PWAV woher))
aus-Dat-*-*-* (QP (PWAV woraus))

außer-Dat-Person-*-* (QP (APPR außer) (PWS wem))

bei-Dat-*-*-* (QP (PWAV wobei))
bei-Dat-Location-*-* (QP (PWAV wo))
bei-Dat-Object-*-* (QP (PWAV wo))
bei-Dat-Person-*-* (QP (APPR bei) (PWS wem))

entgegen-Dat-Person-*-* (QP (PWS wem) (APPO entgegen))

#entsprechend

gegenüber-Dat-Person-*-* (QP (PWS wem) (APPO gegenüber))
```

```
gegenüber-Dat-*-*-* (QP (APPRO gegenüber) (PWS was))

gemäß-Dat-*-*-* (QP (APPRO gemäß) (PWS was))
gemäß-Dat-Person-*-* (QP (APPRO gemäß) (PWS wem))

mit-Dat-Person-*-* (QP (APPRO mit) (PWS wem))
mit-Dat-*-*-* (QP (PWAV womit))

#mitsamt-Dat
#samt-Dat

nach-Dat-Time-MO-* (QP (PWAV wann))
nach-Dat-Person-*-* (QP (APPR nach) (PWS wem))
nach-Dat-Object-MO-* (QP (PWAV wohin))
nach-Dat-Location-MO-* (QP (PWAV wohin))
nach-Dat-*-*-* (QP (PWAV wonach))

nahe-Dat-*-*-* (QP (PWAV wo))

seit-Dat-*-*-* (QP (APPR seit) (PWAV wann))

von-Dat-Person-*-* (QP (APPR von) (PWS wem))
von-Dat-*-*-* (QP (PWAV wovon))
vom-Dat-Person-*-* (QP (APPR von) (PWS wem))
vom-Dat-*-*-* (QP (PWAV wovon))

zu-Dat-Person-*-* (QP (APPR zu) (PWS wem))
zu-Dat-Time-*-* (QP (PWAV wann))
zu-Dat-Location-MO-* (QP (PWAV wohin))
zu-Dat-Object-MO-* (QP (PWAV wohin))
zu-Dat-*-*-* (QP (PWAV wozu))

# dative/genitive prepositions
dank-Dat-Person-*-* (QP (APPR dank) (PWS wem))
dank-Dat-*-*-* (QP (APPR dank) (PWS was))
dank-Gen-*-*-* (QP (APPR dank) (PWS wessen))

laut-Dat-Person-*-* (QP (APPR laut) (PWS wem))
```

```
laut-Dat-*-*-* (QP (APPR laut) (PWS was))
laut-Gen-*-*-* (QP (APPR laut) (PWS was))

# accusative prepositions
bis-Acc-Location-*-* (QP (APPR bis) (PWAV wohin))
bis-Acc-Time-*-* (QP (APPR bis) (PWAV wann))

durch-Acc-Person-*-* (QP (APPR durch) (PWS wen))
durch-Acc-*-*-* (QP (PWAV wodurch))

für-Acc-Person-*-* (QP (APPR für) (PWS wen))
für-Acc-Time-*-* (QP (PWAV wie) (ADJD lange))
für-Acc-*-*-* (QP (PWAV wofür))

gegen-Acc-Person-*-* (QP (APPR gegen) (PWS wen))
gegen-Acc-*-*-* (QP (PWAV wogegen))

#je

ohne-Acc-Person-*-* (QP (APPR ohne) (PWS wen))
ohne-Acc-*-*-* (QP (APPR ohne) (PWS was))

um-Acc-Person-*-* (QP (APPR um) (PWS wen))
um-Acc-*-*-* (QP (APPR um) (PWS was))

wider-Acc-Person-*-* (QP (APPR gegen) (PWS wen))
wider-Acc-*-*-* (QP (PWAV wogegen))
```