Automatic Phrase Recognition in Historical German

Katrin Ortmann
Department of Linguistics
Fakultät für Philologie
Ruhr-Universität Bochum
ortmann@linguistics.rub.de

Abstract

Due to a lack of annotated data, theories of historical syntax are often based on very small, manually compiled data sets. To enable the empirical evaluation of existing hypotheses, the present study explores the automatic recognition of phrases in historical German. Using modern and historical treebanks, training data for a neural sequence labeling tool and a probabilistic parser is created, and both methods are compared on a variety of data sets. The evaluation shows that the unlexicalized parser outperforms the sequence labeling approach, achieving F_1-scores of 87%–91% on modern German and between 73% and 85% on different historical corpora. An error analysis indicates that accuracy decreases especially for longer phrases, but most of the errors concern incorrect phrase boundaries, suggesting further potential for improvement.

1 Introduction

In recent years, the availability of ever-larger data sets and increasing computational power have led to major changes in the way language is analyzed. Today, NLP tools can automatically enrich large amounts of text quickly and accurately with linguistic annotations needed for commercial or research purposes. When it comes to non-standard data like historical language, though, the availability of models and annotated corpora is still limited compared to modern language and hypotheses are often based on qualitative analyses of very small data sets. For example, Speyer (2011) investigates object order in the middle field of Early New High German sentences based on a total of 70 pairs of direct and indirect objects from three centuries. Similarly, Light (2012) grounds her study of extraposition, i.e. the movement of elements behind the clause-final verb, on 115 cases of extraposed subjects in one Early New High German bible translation, while Sapp (2014) analyzes 683 extraposed phrases spread over texts from five centuries. Although data-driven qualitative analyses like these provide valuable insights for linguistic research, they require a lot of manual effort and cannot achieve the same statistical significance as studies of modern language.

Recently, there have been several attempts to address the lack of annotated historical data and provide a basis for the empirical evaluation of existing hypotheses by automatically identifying relevant syntactic units in historical text (e.g. Chiarcos et al., 2018; Ortmann, 2020, 2021). The present paper takes a similar approach and looks explicitly at the units targeted by the qualitative studies mentioned above, namely phrases.

In the context of this study, phrases are understood as continuous, non-overlapping constituents from a sentence's parse tree. Since the concrete definition of constituents may vary depending on the annotation scheme and not all constituents are equally relevant for linguistic studies like the ones mentioned above, this paper focuses on four main phrase types: noun phrases (`NP`), prepositional phrases (`PP`), adjective phrases (`AP`), and adverb phrases (`ADVP`). For each sentence, only the highest non-terminal nodes of the given types are considered, ignoring the internal structure of phrases. This means that phrases may dominate other phrases of the same or different types, but the dominated phrases are not evaluated here. Example (1) shows an annotated sentence from a 1731 theological text.

(1) [$_{\text{NP}}$ Der kraͤftigſte Bewegungs-Grund] nimmt [$_{\text{NP}}$ seinen Urſprung] [$_{\text{PP}}$ aus einer zaͤrtlichen Leydenſchaft meines Gemuͤhts].

The most powerful motive takes its origin from a tender passion of my heart.

To enable research on phenomena like extraposition, phrases may not cross topological field boundaries.[1] For example, a prepositional phrase in the middle field is considered separate from an adjacent modifying relative clause in the post-field, as shown in example (2) from a chemistry essay (field boundaries are indicated by vertical pipes). Also, discontinuous structures as they exist in some German corpora are not allowed here.

(2) Erhebt | [$_{\text{NP}}$ es] [$_{\text{NP}}$ ſich] [$_{\text{PP}}$ mit dem Waſſerſtoffgas], | [$_{\text{NP}}$ welches] | [$_{\text{NP}}$ die Moraͤſte] [$_{\text{PP}}$ in Ueberfluß] | ausdunſten?
Does it rise with the hydrogen gas that the swamps evaporate in abundance?

The goal of this study is to automatically recognize phrases that meet the aforementioned requirements in historical German texts. The remainder of the paper is structured as follows: Section 2 presents related work on the syntactic analysis of (historical) German before Section 3 introduces the data sets used in this study. In Section 4, two different methods for the automatic recognition of phrases are selected based on the findings of previous studies and their performance is evaluated in Section 5. The paper concludes with a discussion in Section 6.

2 Related Work

The recognition of phrases as defined in the previous section is related to chunking as well as (constituency) parsing and can be located somewhere in between the two tasks regarding its complexity.

Chunking refers to the identification of non-overlapping, non-recursive phrases from a sentence's parse tree, ending with the head token (Sang and Buchholz, 2000). As a consequence, chunks are often shorter than phrases because post-modifying elements form separate chunks. For simple cases without pre- or post-modifying elements, however, the definitions of chunks and phrases overlap and methods that are successful at chunking may also be useful for phrase recognition.

Parsing, on the other hand, aims at a complete syntactic analysis of the sentence. Hence, the resulting constituency tree includes more information than just the phrase annotation, e.g. dominance relations, which are not considered in this study. As a result, phrase annotations can be derived from the more complex parse output, but the complexity of the task may also reduce overall accuracy.

While studies on chunking observe F_1-scores >95% for modern German (cf. Müller, 2005; Ortmann, 2021), the highest F_1-scores for constituency parsing of German are reported with approx. 90%, compared to 95% for English (Kitaev et al., 2019). In general, parsing results heavily depend on the selected treebank and the inclusion of grammatical functions (Dakota and Kübler, 2017) and discontinuous structures (cf. Vilares and Gómez-Rodríguez, 2020). Also, all of these results are obtained for standard language like newspaper text. For non-standard data, performance drops must be expected (Pinto et al., 2016; Jamshid Lou et al., 2019).

For historical German, so far, there have been experiments on chunking (Petran, 2012; Ortmann, 2021) and topological field parsing (Chiarcos et al., 2018; Ortmann, 2020). For chunking, the best results are observed for CRF-based sequence labeling with overall F_1-scores between 90% and 94% (Ortmann, 2021). For topological field identification, the application of a probabilistic parser yields overall F_1-scores >92% (Ortmann, 2020). In the present study, both of these approaches will be explored for the purpose of phrase recognition in historical German.

3 Data

The data sets for the experiments are taken from a previous chunking study (Ortmann, 2021).[2] The training data consists of two modern and two historical treebanks. The TüBa-D/Z corpus (Telljohann et al., 2017)[3] and the Tiger corpus (Brants et al., 2004)[4] contain modern German newspaper articles, whereas the Mercurius corpus (Demske, 2005)[5] and the ReF.UP corpus (Demske, 2019)[6] comprise Early New High German texts from the 14^{th} to 17^{th} century. All four data sets are annotated with constituency trees, but before they can be used to train a parser or extract phrase annotations for sequence labeling, a few modifications are necessary.

[1] For an overview of the topological field model, see e.g. Cheung and Penn (2009) or Wöllstein (2018, in German)

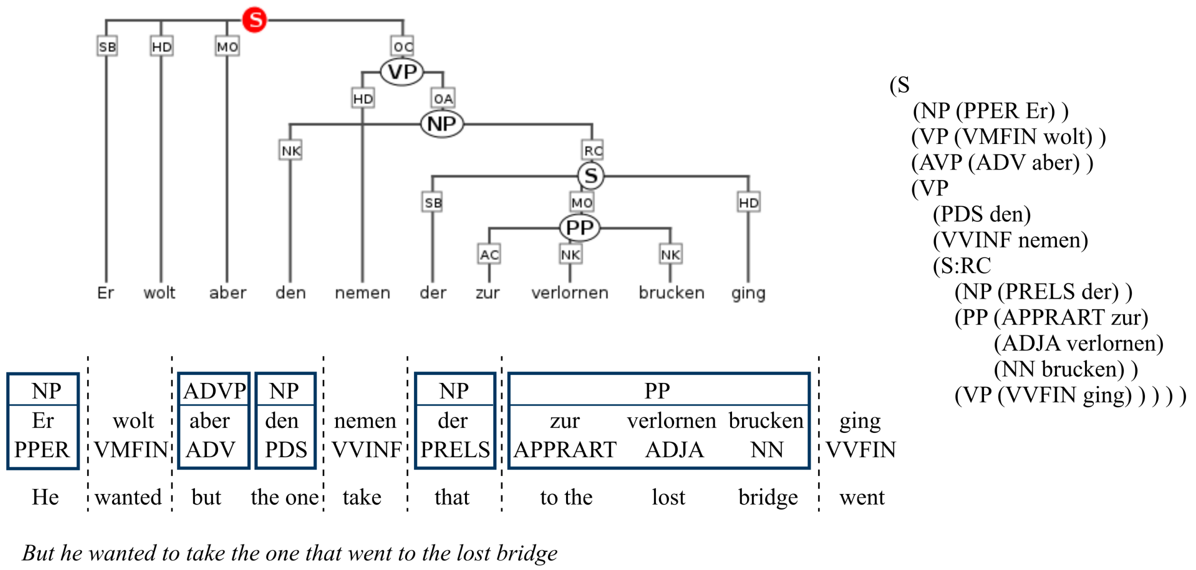
[2] https://github.com/rubcompling/nodalida2021

[3] Release 11.0, http://www.sfs.uni-tuebingen.de/ascl/ressourcen/corpora/tueba-dz.html

[4] Version 2.2, https://www.ims.uni-stuttgart.de/forschung/ressourcen/korpora/tiger

[5] Mercurius Baumbank (version 1.1), https://doi.org/10.34644/laudatio-dev-VyQiCnMB7CArCQ9CjF3O

[6] ReF.UP is a subcorpus of the Reference Corpus of Early New High German (Wegera et al., 2021), https://www.linguistics.rub.de/ref



```
(S
  (NP (PPER Er) )
  (VP (VMFIN wolt) )
  (AVP (ADV aber) )
  (VP
    (PDS den)
    (VVINF nemen)
    (S:RC
      (NP (PRELS der) )
      (PP (APPRART zur)
          (ADJA verlornen)
          (NN brucken) )
      (VP (VVFIN ging) ) ) ) )
```

But he wanted to take the one that went to the lost bridge

Figure 1: Example modification of a sentence from the ReF.UP corpus. At the top, the original constituency tree with discontinuous annotations according to the Tiger scheme is displayed. The bracket structure to the right represents the linearized version of the tree without crossing branches and grammatical functions. This format can be used to train a standard parser. At the bottom, the phrase annotation for the sentence is shown. The phrases have been extracted from the tree structure to the right and checked with a topological field parser to ensure that phrases do not cross field boundaries (indicated by dashed lines). The phrase annotations serve as training data for a sequence labeling tool and are also used for evaluation.

(i) The underlying annotation scheme of the Tiger corpus and the two historical treebanks allows for discontinuous annotations, which must be removed to enable the use of standard chunking and parsing methods. Here, a combination of the raising and splitting approaches described by Hsu (2010) is applied to the trees until no crossing branches remain.[7]

(ii) Since German exhibits a relatively free word order, grammatical functions like subject and object play an important role in the syntactic analysis of sentences, especially for the reduction of ambiguity (Fraser et al., 2013). For the purpose of phrase recognition, however, they are not relevant and, therefore, mostly excluded from the trees to reduce the size of the tagset and improve parsing performance (Rafferty and Manning, 2008; Dakota and Kübler, 2017).[8]

The modified trees can serve as training input for a parser, or they can be used to extract phrase annotations. Contrary to chunking studies, where the lowest non-terminal nodes are converted to chunks (Kübler et al., 2010; Ortmann, 2021), here, the highest non-terminal nodes of the relevant types correspond to the desired phrases.[9] Before the extracted phrases can be used for evaluation or to train a sequence labeling tool, another difference between the annotation schemes of the treebanks regarding topological fields must be taken into account, though.

(iii) While the TüBa-D/Z trees represent a combination of constituency and topological field annotations, the other three corpora that follow the Tiger scheme do not include topological fields. This means that constituents in the TüBa-D/Z data are already bound to the corresponding fields as required by the phrase

[7] Basically, discontinuous nodes are split and re-inserted into the tree based on the linear order of tokens in the sentence. The same holds for punctuation, which is appended to the same parent node as the next token to the left (or to the right for sentence-initial punctuation).

[8] The only exception are GFs that are needed to extract correct phrases from the trees. For the Tiger scheme, these are S:RC and S:OC. For TüBa-D/Z, the following GFs are preserved: KONJ, OS, R-SIMPX, NX:HD within PX, and NX:APP within NX. Also, one-word children of sentence nodes that only receive a grammatical function according to the Tiger scheme are assigned a phrase type `NP`, `PP`, `AP`, `AVP`, `VP`, or `SVP` based on their POS tag.

[9] Again, phrases of the four types are added for one-word constituents from Tiger-scheme trees based on the POS tag of the word.

	News1	News2	Hist	Mix
#Docs	3,075	1,863	28	1,891
#Sents	83,515	40,037	23,747	63,784
#Toks	1,566,250	727,011	569,854	1,296,865
#Phrases	388,531	162,336	152,866	315,202

Table 1: Overview of the four training sets. Only sentences with a gold parse are included, and the number of phrases refers to phrases of the four relevant types. The Mix set is a combination of the News2 and Hist sets.

Corpus	**#Docs**	**#Sents**	**#Toks**	**#Phrases**
TüBa-D/Z	364	10,488	196,630	49,329
Tiger	200	4,445	78,018	17,622
Modern	78	547	7,605	2,240
Mercurius	2	818	18,740	4,401
ReF.UP	26	2,173	54,005	15,355
HIPKON	53	342	4,210	1,146
DTA	29	608	18,515	4,068

Table 2: Overview of the test data. The number of phrases includes NP, PP, AP, and ADVP phrases as described in Section 1. Only sentences containing at least one of the four phrase types are considered.

definition in this study, whereas constituents in the other data sets may cross field boundaries. Therefore, phrases that are extracted from these data sets or identified by a parser that is trained on them are corrected with the help of a topological field parser (Ortmann, 2020).[10] Phrases that cross fields are split at the field boundary and replaced by the dominated sub-phrases to ensure that no phrase is located in more than one field.[11]

An example of the different modifications of the trees and extracted phrases can be found in Figure 1. The resulting data sets are used to build four distinct training sets: News1 corresponds to the TüBa-D/Z data, News2 is based on the Tiger treebank, Hist contains the historical data, and a joined set Mix includes all data sets that follow the Tiger annotation scheme. Table 1 gives a summary of the four training sets.

For evaluation, the test sections of the four treebanks[12] are processed in the same way as the training data, and phrases of the four types are extracted and split at topological field boundaries if necessary. In addition, the chunking study (Ortmann, 2021) provides three other test sets, which were annotated with phrases for the present paper: a corpus of modern non-newspaper data with texts from different registers and two historical data sets from the HIPKON corpus (Coniglio et al., 2014) and the German Text Archive DTA (BBAW, 2021) covering different genres and time periods. Table 2 gives an overview of the test data.[13]

Corpus	NP	PP	AP	ADVP
TüBa-D/Z	54.30	22.47	6.41	16.82
Tiger	55.28	27.55	6.09	11.07
Modern	61.88	17.72	5.94	14.46
Mercurius	50.44	26.68	5.23	17.66
ReF.UP	56.46	20.48	6.11	16.96
HIPKON	51.83	27.40	2.01	18.76
DTA	51.55	25.76	6.15	16.54

Table 3: Distribution of the four phrase types in the test data. Numbers are given in percent.

In Table 3, the distribution of the phrase types in the data sets is displayed. The most frequent phrase type are NPs with 50% to over 60% in the modern non-newspaper data, followed by PPs with 18% to 28%. ADVPs make up for 11% to 19%, while APs that are not dominated by other phrases are rare with 6% or less.

4 Methods

So far, the automatic syntactic analysis of historical German has been focused on the identification of chunks and topological fields. As described in Section 2, the best results for these tasks are reported for sequence labeling and statistical parsing. In the following, both approaches are applied to the recognition of phrases.

For sequence labeling, the neural CRF-based sequence labeling tool NCRF++ (Yang and Zhang, 2018) is selected. It achieves state-of-the-art performance for several tasks, including tagging, chunking, and named entity recognition in English (Yang et al., 2018). When POS tags are used as features, it also proves successful at identifying chunks in historical German with F_1-scores >90% (Ortmann, 2021). The default configuration consists of a three-layer architecture with a character and a word se-

[10] https://github.com/rubcompling/latech2020

[11] Theoretically, it would also be possible to merge the constituency trees with automatically created topological field annotations before training a parser on the merged trees. However, experiments indicate that this creates too many inconsistencies in the training data, e.g. due to errors in the automatic field annotation, and therefore leads to worse results than splitting the extracted phrase output at the field boundaries afterwards.

[12] While the Tiger corpus is provided with official training, development, and test sections, for the other three corpora, the same splits into training (80%), development (10%), and test set (10%) as in the chunking study (Ortmann, 2021) are used.

[13] The manually annotated data sets can be found in this paper's repository at https://github.com/rubcompling/konvens2021.

quence layer plus a CRF-based inference layer. For the present study, the toolkit is trained on the extracted phrases from the four training sets, where phrases are represented as BIO tags. POS tags are included as additional feature and, during training, the tool is also provided with the development sections of the training corpora. For every word, NCRF++ outputs the single most likely BIO tag, i.e. `B-XP` (beginning of phrase), `I-XP` (inside of phrase), or `O` (outside of phrase). For evaluation, the labels are converted to phrases, and the best result over five runs with different random seeds is reported.

For parsing, the unlexicalized Berkeley parser (Petrov et al., 2006)[14] is selected. It achieves a parsing F_1-score of 91.8% on the TüBa-D/Z corpus and 72% on the Tiger corpus (Dakota and Kübler, 2017) and has also been successfully applied to topological field parsing of historical German with overall F_1-scores >92% (Ortmann, 2020). In the present study, it is trained with default settings[15] on the four training sets, where the modified constituency trees are used as training input. For annotation, the parser is invoked in interactive mode[16] and given a sentence annotated with POS tags, it returns the single best parse. For evaluation, the constituency trees are then converted to phrases as described in the previous section.

5 Evaluation

To evaluate the performance of the selected approaches on the task of phrase recognition, the output of the trained systems is compared to the gold standard annotation. However, the evaluation of sequence annotations like phrases with standard metrics faces the problem of double penalties, meaning that one unit can count as two errors. For example, and adjective phrase that is recognized as adverb phrase would correspond to a false negative `AP` and, at the same time, a false positive `ADVP`. Similarly, if a system misses the initial preposition of a `PP` and instead annotates the rest as an `NP`, this would result in a false negative `PP` and a false positive `NP`. There have been different suggestions on how to deal with this problem. For word tokenization, Shao et al. (2017) argue that recall should be used as the only evaluation metric. While precision favors under-splitting systems, recall values clearly show the percentage of correctly recognized units that are relevant for higher-level tasks. However, in the case of segmentation tasks that include labeling, identifying entities with almost correct boundaries may also be useful (cf. Ortmann, 2021). For example, the studies on extraposition mentioned in Section 1 would still benefit greatly from the recognition of incomplete phrases, if not for a complete automatic analysis, then at least for an easier and faster compilation of much larger data sets (see also Eckhoff and Berdičevskis (2016) for a study on using automatic dependency parsing for pre-annotation of historical data to speed up manual annotation). Hence, precision values should not be disregarded entirely. Instead, in Ortmann (2021), I proposed a more fine-grained error analysis that takes into account different types of possible errors while at the same time circumventing the problem of multiply penalizing errors in a single unit.

In the following, this error analysis is adopted for the evaluation of phrase recognition and the output of the different methods and models is compared phrase-wise to the gold standard annotation, grouping phrases into one of seven classes: true positives (`TP`), false positives (`FP`), labeling errors (`LE`), boundary errors (`BE`), labeling-boundary errors (`LBE`) and false negatives (`FN`). In addition to the standard categories, labeling errors refer to phrases that cover the same token span but are labeled with a different phrase type. Boundary errors are phrases of the correct type but with incorrect boundaries, and labeling-boundary errors are a combination of the former two error types. Since the three error types indicate an existing and not a missing annotation, they are counted as false positives for the calculation of F-scores. Only sentences containing at least one of the four phrase types are evaluated, and punctuation at phrase boundaries is ignored.

Sequence labeling As already mentioned, the neural sequence labeling tool NCRF++ has been applied successfully to the identification of chunks in German, reaching F_1-scores between 90% and 94% for different historical data sets (Ortmann, 2021). As could be expected from previous studies (e.g., Petran, 2012), the accuracy for the recognition of phrases, i.e. longer units, with CRF-based sequence labeling is considerably lower. Table 4 gives a sum-

[14] https://github.com/slavpetrov/berkeleyparser

[15]
```
java -cp BerkeleyParser-1.7.jar
edu.berkeley.nlp.PCFGLA.GrammarTrainer
-treebank SINGLEFILE -out grammar.gr
-path treebank.txt
```

[16]
```
java -jar BerkeleyParser-1.7.jar -gr
grammar.gr -maxLength 1000 -useGoldPOS
```

Corpus	`News1`	`News2`	`Hist`	`Mix`
TüBa-D/Z	**85.18**	76.82	n.a.	n.a.
Tiger	78.93	**79.69**	n.a.	n.a.
Modern	**86.80**	83.10	n.a.	n.a.
Mercurius	**70.25**	67.83	9.05	8.93
ReF.UP	**70.62**	67.91	8.80	9.90
HIPKON	80.13	**81.18**	8.17	7.99
DTA	**72.02**	68.89	6.93	7.78

Table 4: Overall F_1-scores of the sequence labeling approach. Models trained on historical data are only applied to the historical test sets. The table reports the highest F_1-score over five runs and the best result for each corpus is highlighted in bold.

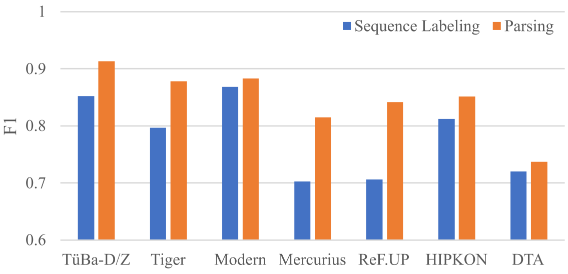


Figure 2: Comparison of the best F_1-scores for sequence labeling and parsing on the different test sets.

mary of the results for each of the four models.

Using gold POS tags as a feature, the two newspaper-based models still perform relatively well. Model `News1` achieves the best results with F_1-scores between 70.3% and 86.8%. The results for the second modern model `News2` also lie above 67% for all data sets. Contrary to the results for chunking (Ortmann, 2021), using historical training data does not improve the results on the historical test sets. Instead, the historical and mixed models do not reach F_1-scores $>$10% for phrase recognition, indicating that the tool was not successful at learning to identify the different phrase types based on the historical corpora. Possible reasons could be the high syntactic complexity of Early New High German sentences or too much variation in the training data, e.g. caused by the non-standardized spelling in historical German.

Parsing So far, the parsing approach has only been evaluated for topological field parsing of historical German with overall F_1-scores $>$92% (Ortmann, 2020). In Table 5, the results of the Berkeley parser for the recognition of phrases are given. On the modern data sets, the parser achieves F_1-scores of 87.8% to 91.3% with visible differences between the two modern models. While, unsurprisingly, each of them performs best on the test section of the corpus it was trained on, the `News1` model also achieves the best results on the Modern data set and the DTA corpus, while the `News2` model performs better on the other historical data sets.

Corpus	`News1`	`News2`	`Hist`	`Mix`
TüBa-D/Z	**91.30**	81.50	n.a.	n.a.
Tiger	82.73	**87.81**	n.a.	n.a.
Modern	**88.27**	84.44	n.a.	n.a.
Mercurius	60.32	65.72	**81.50**	81.06
ReF.UP	56.44	58.86	**84.15**	84.05
HIPKON	74.44	75.13	85.05	**85.12**
DTA	**73.66**	69.44	69.07	70.63

Table 5: Overall F_1-scores (in percent) for the four parser models on each data set. Models trained on historical data are only applied to the historical test sets, and the highest F_1-score for each corpus is highlighted in bold.

In contrast to the sequence labeling results, here, including historical training data improves the syntactic analysis of historical language, probably because the unlexicalized parser is unaffected by the non-standardized spelling or can better handle the complex sentence structures. For three of the four historical data sets, the `Hist` and `Mix` models outperform the modern models by ten percentage points or more. F_1-scores lie between 81.5% and 85.1% for the Mercurius, ReF.UP and HIPKON data, while the DTA is only analyzed with an F_1-score of 73.7%.

When compared to the sequence labeling tool, the parsing approach yields better results for the recognition of phrases. Figure 2 confirms that the best parser model outperforms the best sequence labeling model by up to 13.5 percentage points on each data set. Only for the modern non-newspaper data and the DTA, the results of the methods are similar. For the modern data, this could be due to the fact that the data set contains many non-complex phrases that are similar to chunks, e.g. simple noun phrases. 54% of the phrases in this data set consist of only one token, compared to 35%–50% in the other data sets, which makes it easier for the sequence labeling approach to identify them.

However, parser accuracy also declines for larger units (cf. Bastings and Sima'an, 2014). While the Berkeley parser reaches overall parsing F_1-scores of 92% and 86% for the modern data and 78%–79% for the historical data (cf. Table 6), F_1-scores heavily decline for larger constituents as well as phrases (see Figure 3). For constituents with more than five words, the average F_1-score of the four mod-

	News1	News2	Hist	Mix
TüBa-D/Z	**91.96**	n.a.	n.a.	n.a.
Tiger	n.a.	**86.42**	n.a.	n.a.
Mercurius	n.a.	52.27	**77.68**	77.44
ReF.UP	n.a.	45.15	78.97	**79.13**

Table 6: Overall labeled F_1-score for the four trained parser models on the test data, excluding virtual root nodes. Training and test trees are modified as described in Section 3, and models are only evaluated on test data that follows the same syntactic annotation scheme as the training data.

Corpus	NP	PP	AP	ADVP
TüBa-D/Z	89.03	83.26	86.99	91.40
Tiger	86.60	79.28	75.80	82.35
Modern	87.35	76.37	80.60	79.94
Mercurius	77.96	70.47	62.61	82.59
ReF.UP	82.72	75.21	63.31	81.77
HIPKON	80.49	77.62	60.00	84.49
DTA	66.53	64.98	67.98	72.06

Table 7: Overall F_1-scores for each phrase type (in percent) for the best performing parser model on each data set.

Corpus	FP	LE	BE	LBE	FN
TüBa-D/Z	22.47	0.96	62.85	0.75	12.97
Tiger	20.15	1.08	59.22	1.15	18.41
Modern	19.12	1.99	64.34	0.40	14.14
Mercurius	26.84	1.23	51.94	1.49	18.50
ReF.UP	22.74	1.53	53.20	1.23	21.30
HIPKON	20.00	3.03	66.36	1.21	9.39
DTA	17.73	1.01	60.91	2.47	17.88

Table 8: Proportion of the five error types: false positives (FP), labeling errors (LE), boundary errors (BE), labeling-boundary errors (LBE), and false negatives (FN). Numbers are given in percent for the best parser model on each data set.

els is only about 70%. For phrases, the reduction is even larger with F_1-scores below 40% for phrases of twenty or more words. This observation may, in part, explain the lower results for the DTA because, proportionally, this data set contains about twice as many phrases of twelve or more words than the other corpora due to many dedications and very long phrases with coordinations and dominated sentences, e.g. in legal texts. A parser that performs better on larger constituents thus might be better equipped to analyze this data set.

Table 7 reports the parser results broken down by phrase types. Here, each category is evaluated separately and one unit may thus appear in two categories, e.g. as a false negative PP and a false positive NP as exemplified above. For most data sets, the highest F_1-scores are reached for adverb and noun phrases. While the former are usually very short and therefore easier to identify, noun phrases and prepositional phrases often contain pre- and/or postnominal modifiers including longer constituents like relative clauses that lead to errors in the parser output. Adjective phrases are the least frequent phrase type and, although they tend to be short, also show the least accurate results for more than half of the data sets. Often they get mixed up with neighboring adverbs because a lexicalized model would be necessary to distinguish between pre-modifying adverbs as in example (3) and a separate adverb phrase in (4).

(3) Sie war [$_{\text{AP}}$ sehr/ADV glücklich/ADJD].
She was very happy.

(4) Sie war [$_{\text{ADVP}}$ gestern/ADV] [$_{\text{AP}}$ glücklich/ADJD].
Yesterday, she was happy.

Finally, Table 8 shows the distribution of error types for the best parser models. For all test sets, boundary errors are by far the most frequent error types with a proportion of 52% to 66%. The remaining errors are mostly traditional false positives and false negatives, while labeling and labeling-boundary errors are rare. Considering that the identification of phrases with almost correct boundaries may still satisfy the requirements of certain tasks as discussed above, this can thus be assumed for more than half of the errors. Furthermore, the results suggest great potential for improvement because the high percentage of boundary errors means that the parser already identified these phrases, and correcting boundaries could potentially lead to significant increases in precision.

6 Discussion

The present study has explored the automatic recognition of phrases in historical German. Two tools that proved successful in previous studies on chunking and topological field parsing were selected and trained on modern and historical treebanks. The evaluation has shown that the Berkeley parser outperforms the neural CRF-based sequence labeling tool NCRF++ on all data sets, reaching overall F_1-scores of 87.8% to 91.3% on modern German and 73.7%–85.1% on different historical corpora. Parsing results are most accurate for simple phrases while scores decline with increasing phrase length. Since the majority of errors turn out to be boundary errors, the results leave room for further improve-

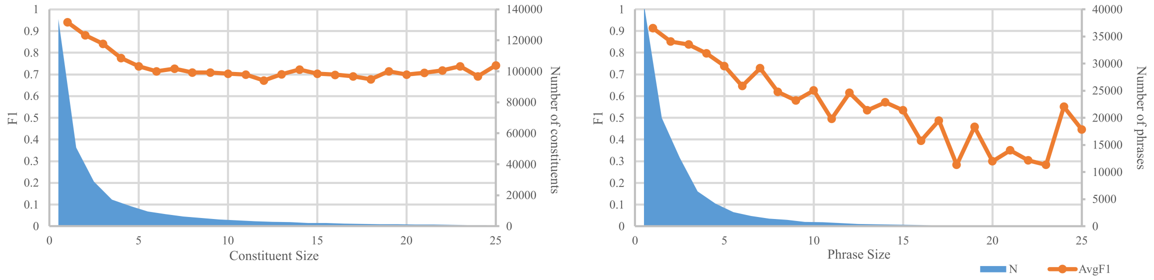


Figure 3: Average F_1-score of the four parser models for the recognition of constituents and phrases of sizes 1–25. The number of constituents includes all constituents of the given sizes in the test sections of the four training corpora. The number of phrases refers to phrases of the four types in the seven test sets.

ment of annotation precision.

Interestingly, the inclusion of historical training data improves the results of the parser, whereas the sequence labeling tool did not benefit from it. One possible explanation could be too much variation in the data due to the non-standardized spelling in historical German, which does not affect the unlexicalized parser. Future studies could experiment with spelling normalization, which was observed to improve the annotation results of modern NLP tools for parsing Middle English (Schneider et al., 2015) or tagging historical German (Bollmann, 2013) and Dutch (Tjong Kim Sang et al., 2017).

The normalized data could then also be used to explore lexicalized parsing, e.g. with the neural Berkeley parser (Kitaev and Klein, 2018). Although parsers do not necessarily need lexical information for good performance (Coavoux et al., 2019), studies on modern English show that the application of neural parsing methods in combination with pre-trained word embeddings can further improve the results (cf. e.g. Vilares and Gómez-Rodríguez, 2020). For morphologically more complex languages like German, this should be even more relevant (Fraser et al., 2013) and could also help in cases where lexical information is necessary to decide about the correct phrase boundaries.

Acknowledgments

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – Project-ID 232722074 – SFB 1102 (Project C6). I am grateful to the student annotators Anna Maria Schröter and Larissa Weber for the annotations. Also, I would like to thank the anonymous reviewers for their helpful comments.

References

Joost Bastings and Khalil Sima'an. 2014. All fragments count in parser evaluation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, pages 78–82, Reykjavik, Iceland. European Languages Resources Association (ELRA).

BBAW. 2021. Deutsches Textarchiv. Grundlage für ein Referenzkorpus der neuhochdeutschen Sprache. Berlin-Brandenburgische Akademie der Wissenschaften.

Marcel Bollmann. 2013. POS tagging for historical texts with sparse training data. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 11–18, Sofia, Bulgaria. Association for Computational Linguistics.

Sabine Brants, Stefanie Dipper, Peter Eisenberg, Silvia Hansen-Schirra, Esther König, Wolfgang Lezius, Christian Rohrer, George Smith, and Hans Uszkoreit. 2004. TIGER: Linguistic interpretation of a German corpus. *Research on language and computation*, 2(4):597–620.

Jackie Chi Kit Cheung and Gerald Penn. 2009. Topological field parsing of German. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - Volume 1*, ACL '09, page 64–72, USA. Association for Computational Linguistics.

Christian Chiarcos, Benjamin Kosmehl, Christian Fäth, and Maria Sukhareva. 2018. Analyzing Middle High German syntax with RDF and SPARQL. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Maximin Coavoux, Benoît Crabbé, and Shay B. Cohen. 2019. Unlexicalized transition-based discontinuous constituency parsing. *Transactions of the Association for Computational Linguistics*, 7:73–89.

Marco Coniglio, Karin Donhauser, and Eva Schlachter. 2014. HIPKON: Historisches Predigtenkorpus zum

Nachfeld (Version 1.0). Humboldt-Universität zu Berlin. SFB 632 Teilprojekt B4.

Daniel Dakota and Sandra Kübler. 2017. Towards replicability in parsing. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 185–194, Varna, Bulgaria. INCOMA Ltd.

Ulrike Demske. 2005. Mercurius-Baumbank (Version 1.1). Universität Potsdam.

Ulrike Demske. 2019. Referenzkorpus Frühneuhochdeutsch: Baumbank.UP. Universität Potsdam.

Hanne Martine Eckhoff and Aleksandrs Berdičevskis. 2016. Automatic parsing as an efficient pre-annotation tool for historical texts. In *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH)*, pages 62–70, Osaka, Japan. The COLING 2016 Organizing Committee.

Alexander Fraser, Helmut Schmid, Richárd Farkas, Renjing Wang, and Hinrich Schütze. 2013. Knowledge sources for constituent parsing of german, a morphologically rich and less-configurational language. *Computational Linguistics*, 39(1):57–85.

Yu-Yin Hsu. 2010. Comparing conversions of discontinuity in pcfg parsing. In *Ninth International Workshop on Treebanks and Linguistic Theories*, pages 103–113.

Paria Jamshid Lou, Yufei Wang, and Mark Johnson. 2019. Neural constituency parsing of speech transcripts. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2756–2765, Minneapolis, Minnesota. Association for Computational Linguistics.

Nikita Kitaev, Steven Cao, and Dan Klein. 2019. Multilingual constituency parsing with self-attention and pre-training. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3499–3505, Florence, Italy. Association for Computational Linguistics.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Melbourne, Australia.

Sandra Kübler, Kathrin Beck, Erhard Hinrichs, and Heike Telljohann. 2010. Chunking German: an unsolved problem. In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 147–151, Uppsala, Sweden. Association for Computational Linguistics.

Caitlin Light. 2012. The information structure of subject extraposition in early new high german. *University of Pennsylvania Working Papers in Linguistics*, 18(1):20.

Frank Henrik Müller. 2005. *A finite-state approach to shallow parsing and grammatical functions annotation of German*. Ph.D. thesis, Seminar für Sprachwissenschaft, Universität Tübingen.

Katrin Ortmann. 2020. Automatic Topological Field Identification in (Historical) German Texts. In *Proceedings of the The 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 10–18.

Katrin Ortmann. 2021. Chunking Historical German. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 190–199.

Florian Petran. 2012. Studies for segmentation of historical texts: Sentences or chunks? In *Proceedings of the Second Workshop on Annotation of Corpora for Research in the Humanities (ACRH-2)*, pages 75–86.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 433–440.

Alexandre Pinto, Hugo Gonçalo Oliveira, and Ana Oliveira Alves. 2016. Comparing the performance of different NLP toolkits in formal and social media text. In *5th Symposium on Languages, Applications and Technologies (SLATE'16)*, pages 3:1–3:16. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

Anna N Rafferty and Christopher D Manning. 2008. Parsing three german treebanks: Lexicalized and unlexicalized baselines. In *Proceedings of the Workshop on Parsing German*, pages 40–46.

Erik F. Tjong Kim Sang and Sabine Buchholz. 2000. Introduction to the CoNLL-2000 shared task: Chunking. In *Fourth Conference on Computational Natural Language Learning and the Second Learning Language in Logic Workshop*, pages 127–132.

Christopher D. Sapp. 2014. Extraposition in middle and new high german. *The Journal of Comparative Germanic Linguistics*, 17(2):129–156.

Gerold Schneider, Hans Martin Lehmann, and Peter Schneider. 2015. Parsing early and late modern english corpora. *Literary and Linguistic Computing*, 30(3):423–439.

Yan Shao, Christian Hardmeier, and Joakim Nivre. 2017. Recall is the proper evaluation metric for word segmentation. In *Proceedings of the The 8th International Joint Conference on Natural Language Processing*, pages 86–90, Taipei, Taiwan.

Augustin Speyer. 2011. Die Freiheit der Mittelfeldabfolge im Deutschen. Ein modernes Phänomen. *Beiträge zur Geschichte der deutschen Sprache und Literatur*, 133(1):14–31.

Heike Telljohann, Erhard W. Hinrichs, Sandra Kübler, Heike Zinsmeister, and Kathrin Beck. 2017. *Stylebook for the Tübingen Treebank of Written German (TüBa-D/Z)*. Seminar für Sprachwissenschaft, Universität Tübingen, Germany.

Erik Tjong Kim Sang, Marcel Bollman, Remko Boschker, Francisco Casacuberta, FM Dietz, Stefanie Dipper, Miguel Domingo, Rob van der Goot, JM van Koppen, Nikola Ljubešić, et al. 2017. The clin27 shared task: Translating historical text to contemporary language for improving automatic linguistic annotation. *Computational Linguistics in the Netherlands Journal*, 7:53–64.

David Vilares and Carlos Gómez-Rodríguez. 2020. Discontinuous constituent parsing as sequence labeling. *arXiv preprint arXiv:2010.00633*.

Klaus-Peter Wegera, Hans-Joachim Solms, Ulrike Demske, and Stefanie Dipper. 2021. Referenzkorpus Frühneuhochdeutsch (Version 1.0).

Angelika Wöllstein. 2018. Topologisches Satzmodell. In Jörg Hagemann and Sven Staffeldt, editors, *Syntaxtheorien. Analysen im Vergleich*, 2., aktualisierte auflage edition, pages 145 – 166. Stauffenburg, Tübingen.

Jie Yang, Shuailong Liang, and Yue Zhang. 2018. Design challenges and misconceptions in neural sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*, pages 3879–3889, Santa Fe, New Mexico, USA.

Jie Yang and Yue Zhang. 2018. NCRF++: An open-source neural sequence labeling toolkit. In *Proceedings of ACL 2018, System Demonstrations*, pages 74–79, Melbourne, Australia.