German Parliamentary Corpus (GerParCor) Reloaded

Giuseppe Abrami, Mevlüt Bagci, Alexander Mehler
Goethe University Frankfurt, Text Technology Lab
Robert-Mayer-Straße 10, 60325 Frankfurt am Main
{abrami, bagci, mehler}@em.uni-frankfurt.de

Abstract

In 2022, the largest German-speaking corpus of parliamentary protocols from three different centuries, on a national and federal level from the countries of Germany, Austria, Switzerland and Liechtenstein, was collected and published – GerParCor. Through GerParCor, it became possible to provide for the first time various parliamentary protocols which were not available digitally and, moreover, could not be retrieved and processed in a uniform manner. Furthermore, GerParCor was additionally preprocessed using NLP methods and made available in XMI format. In this paper, GerParCor is significantly updated by including all new parliamentary protocols in the corpus, as well as adding and preprocessing further parliamentary protocols previously not covered, so that a period up to 1797 is now covered. Besides the integration of a new, state-of-the-art and appropriate NLP preprocessing for the handling of large text corpora, this update also provides an overview of the further reuse of GerParCor by presenting various provisioning capabilities such as API's, among others.

Keywords: Parliament protocol, German, Corpus, UIMA, DUUI

1. Introduction

In 2022, the largest German-language corpus of parliamentary protocols from three different centuries along for national and federal levels from Germany, Austria, Switzerland, and the Principality of Liechtenstein has been published at this time – GerParCor (Abrami et al., 2022). GerParCor includes all protocols online available as well as a number of full legislative sessions that are not available online. Furthermore, in addition to providing the protocols, they were also linguistically annotated using *spaCy* (Honnibal et al., 2020) with the help of TextImager (Hemati et al., 2016). This Natural Language Processing (NLP) task, carried out using TextImager, produces annotations based on UIMA (*Unstructured Information Management applications* Ferrucci et al. (2009)) and allows a structured reuse of the documents.

The parliamentary protocols, as well as the other parliamentary documents (printed matters, parliamentary questions and answers) accumulate a growing treasure (e.g. Bornheim et al. (2023)). However, access to them is hampered by the heterogeneity of the data access points of the individual national and federal parliaments. Thus, in this paper, we updated GerParCor and extensively expanded this corpus.

These corpus upgrades include the incorporation of new parliamentary protocols in conjunction with NLP preprocessing using a novel, faster and more scalable NLP framework named Docker Unified UIMA Interface (DUUI– Leonhardt et al. (2023)), as well as additional technical features for securing reusability of the parliamentary protocols. In this regard, the paper is organized as follows: In Section 2, the existing German parliamentary corpora and their enhancements since the last publication are addressed before the features of GerParCor, introduced with this paper, are described in Section 3. After that, the novel preprocessing of GerParCor and the resulting formats are described in Section 4. Section 5 documents possibilities for post-utilization of GerParCor. In Section 6, we provide statistical information about the new release followed by an outlook on future work in Section 7. Section 8 concludes the paper.

2. Related Work

The landscape for German-language parliamentary corpora has not changed significantly over the past year. There are some projects that offer multilingual corpora (e.g. *ParlSpeech V2* (Rauh and Schwalbach, 2020), *ParlaMint* (Erjavec et al., 2022)), but we focus on pure German corpora, because the multilingual ones have only a small part of them. Within Truan and Romary (2021)'s corpus, only the federal parliamentary debates between 1998 and 2015 are identified, wihel with Barbaresi (2018)'s work, only the speeches of the 200 most important political actors in the period between 1982 and 2020 are identified. Likewise, only the parliamentary protocols of the German Bundestag (in the period 1949 to the present) can be found in *GermaParl* (Blaette and Leonhardt, 2023). In addition, *ParlAT beta* (Wissik and Pirker, 2018) offers an accumulation of parliamentary debates from the Austrian parliamentary National Chamber in the period 1996 - 2017. Another equally small corpus is the Austrian corpus of Sippl et al. (2016)

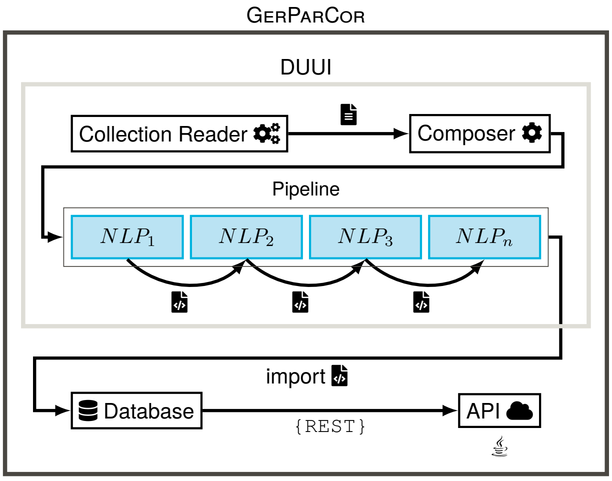

Figure 1: GerParCor encapsulates DUUI for automatic processing of parliamentary protocols. In this process, all parliamentary protocols are read by the *Collection Reader* and sent through a defined pipeline by the *Composer*. At each step, the processed protocols are enriched with annotations, which are finally stored in a database. The parliamentary protocols can be accessed using the provided Java API.

including the years 2013 – 2015 and annotated with the Stanford Tagger (Toutanvoa and Manning, 2000).

To the best of our knowledge, the longest period of availability of German parliamentary corpora is currently covered by GerParCor, which will be considerably extended by this work.

3. GerParCor Updates

The update of GerParCor includes the following features:

1. The convolutions of the already existing parliaments have been continued since the last update.

2. In the last version, parliamentary debates from earlier chronological periods were not yet available in digital version. With this update, these are also available.

3. In the previous version, only the German-speaking regional parliaments from Germany were included. In this update, the regional parliaments from Austria are now also included (see Table 2).

4. GerParCor now includes other historical parliamentary protocols from regional parliaments, which span a period back to 1797 (see Table 3).

5. Due to the size of the corpus and the amount of text, we switched the NLP preprocessing from TextImager to DUUI, which allows us to annotate text more efficiently. In this context, we have also added further NLP preprocessing steps that extend the existing spaCy-based annotations. Moreover, DUUI components (i.e. NLP methods) have been implemented and added, which all are reusable via DUUI (see Section 4).

6. After receiving many requests to reuse GerParCor, which is currently fully available as a UIMA export in XMI format, we learned that UIMA is considered a complicated format. We have therefore decided to offer other export formats such as CoNLL as well as plain text.

7. For more dynamic access to the corpus, we provide a Java API to browse and selectively download single or multiple sessions in a particular format. For this purpose, a UIMA database (the so called GerParCor database) was created using MongoDB[1], which stores all plenary protocols and their metadata by reusing the results of Abrami and Mehler (2018).

8. Finally, we have fundamentally changed the web interface for GerParCor, which is now based on the new GerParCor database, meaning that it becomes possible to navigate across the web interface, get an overview of the corpus and download individual plenary protocols in the required data format. All other operations can be performed using the API mentioned above.

9. Besides the API, the usage of DUUI enables us to provide a DUUI reader for GerParCor, which avoids the intermediary step of downloading the whole or a part of the corpus to the local system. In this way, all operations can be processed directly within UIMA.

As this update is based on the previous version, the latter version will be fully preserved and will remain usable as part of the revised version of the website. GerParCor's update includes a 26,3% increase for sentences and a 28,35% increase for tokens. During the update of GerParCor it became apparent how differently the individual parliaments – regardless of whether regional or national – have structured and organized their parliamentary documentation. These range from direct download links per election period of a parliamentary chamber to complex information portals,

[1] https://www.mongodb.com

Feature	Reference
Token	Honnibal et al. (2020)
Sentence	
Part-Of-Speech	
Lemma	
Named Entity	
Dependency	
Sentiment	Tymann et al. (2019)

Table 1: Overview of the annotated linguistic features and the used resources which were implemented using DUUI

which in the latter case lead to a considerable increase in individual effort for the systematic compilation of GerParCor. While some parliamentary documentations lead by example and share the common parliamentary documentation[2], this is unfortunately not uniformly the case. Even though this is probably an elaborate undertaking, the timing for a uniform and structured automated input of parliamentary documents by the parliamentary documentation would be desirable, unless there are efforts on a bilateral level to offer a uniform data format in conjunction with a machine-readable interface. Since this effort does not yet exist, GerParCor is the alternative to make the largest German-language parliamentary corpus available to researchers, teachers, students and other interested groups in a uniform and now also interface-based way.

4. NLP Processing

Since GerParCor contains a very large body of texts from different chronological periods and, in addition, the individual documents are very large in total, NLP is an expensive effort in total. Regardless of this, an already annotated corpus is not only an aim of the GerParCor project but also an absolute prerequisite for any further use. As for the existing corpus, the new texts added in GerParCor are processed with *spaCy* to recognize basic linguistic annotations such as, sentences, tokens, part-of-speech, lemmas as well as dependency-annotations and named entities. In addition, we perform a sentiment analysis. The list of annotation tasks performed on GerParCor is shown in Table 1. These NLP operations are performed using DUUI (Leonhardt et al., 2023) a tool designed for UIMA-based processing of large text corpora by means of a container-based component orchestration with the help of Docker. In contrast to TextImager (Hemati et al., 2016), used to process the last version of GerParCor, new tools can be integrated faster using DUUI, whose scalability is higher due to its Docker Swarm (Cluster) capability. Regardless of this, it is also a challenge for DUUI to process very long texts if the individual components have hardware limitations due to the underlying model or processing procedure (see Figure 1). But this challenge is exactly the type of task for which DUUI was developed, and the latest – but yet unpublished – update to DUUI allows large documents to be processed efficiently without the need to customize any of the individual analysis components. This is achieved by performing a document-wise segmentation, which results in smaller document segments to be processed. This approach works very well for a variety of NLP components. Once the segments have been processed, they are merged back into the overall document, with the annotations resulting from the individual pipeline steps being unified.

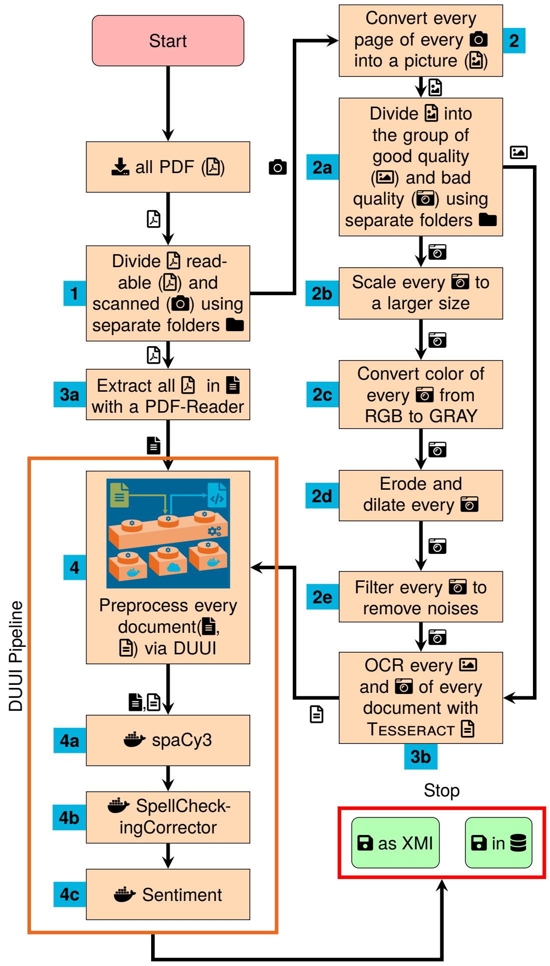
Start
all PDF
1
Divide read-able and scanned using separate folders
3a
Extract all in with a PDF-Reader
DUUI Pipeline
4
Preprocess every document(,) via DUUI
4a
spaCy3
4b
SpellCheck-ingCorrector
4c
Sentiment
Convert every page of every into a picture
2
2a
Divide into the group of good quality and bad quality using separate folders
2b
Scale every to a larger size
2c
Convert color of every from RGB to GRAY
2d
Erode and dilate every
2e
Filter every to remove noises
OCR every and of every document with Tesseract
3b
Stop
as XMI
in

Figure 2: Workflow of GerParCor's OCR process including new NLP preprocessing and OCR filter (adapted from Abrami et al. (2022)). All DUUI processes, which are combined in the pipeline, are available as Docker images.

All annotations performed with DUUI are arranged in a pipeline, which has been extended compared to the previous version. Not only has

[2] https://www.parlamentsspiegel.de

the workflow (see Figure 2) for conversion and NLP analysis of plenary transcripts been migrated to DUUI, but additional filtering functions for the OCR process have been added. We have enhanced the filtering of images for the old data. Further elaboration on this process can be found in the workflow explanation. The updated GerParCor utilizes the DUUI pipeline instead of TextImager. This pipeline incorporates additional preprocessing steps and updated tools for GerParCor. Apart from spaCy3, we have integrated sentiment analysis into the system. Preprocessing with DUUI also incorporates *Symspell*, a spellchecking tool of the Python library *sysmspellpy* (mammothb, 2018). In addition to computing metadata such as *quality* and *word percentages*, we store spelling information, including the corrections made to wrong words (see below). The adapted workflow looks as follows:

1. Initially, we gather all PDFs pertinent to the GerParCor corpus and categorize the readable PDFs and scanned documents into distinct folders.

2. Then we convert the pages of scanned documents (📷) into images (🖻) (python library: pdf2image (Belval, 2017)):

    (a) We classify the images into two classes (of good and poor quality); if there are no scans of poor quality, we proceed to point 3.

    b-d We re-scale to a larger size, colour in grey, and erode and dilate each of the bad scans (using Bradski (2000)).

    e We decided to implement Otsu's binarization filter instead of the adaptive thresholding method of the preview (using Bradski (2000)). The latter, while removing fillings from words, inadvertently compromised the quality of the OCR process.

3. Text-extraction:

    3a We use PDF extractor (python library: textract (Malmgren, 2014)) to extract the text of every readable PDF document (📄).

    3b We use Tesseract to extract the text of the image pages of every scanned document.

4. NLP-Processing via DUUI.

    4a We preprocess each document using all components of spaCy for German.

    4b The Output quality of the scanned documents is measured using *SymSpell*, which also correct wrong words.

    4c Sentence- and document-based sentiment analysis is done using XLM-T (Barbieri et al., 2022).

5. After NLP preprocessing the output is saved. The output options are: XMI or database (🛢).

The UIMA annotation is exemplified in Figure 3. Using *SymSpell*, we process each token identified using spaCy3, verifying its spelling. Note that we exclude all non-alphanumeric tokens, including digits, from this process. In GerParCor, *SymSpell* provides four potential output options (spelling-type):

1. The word is considered correct if the output matches the input word exactly: mark as correct.

2. The word is deemed incorrect if the output does not match the input word: mark as wrong and save the suggestion of *Symspell*.

3. If the output is empty, mark the word as unknown and in such cases, *Symspell* is unable to correct the word.

4. If the input word is non-alphanumeric or a digit, mark the word as skipped.

The previous version of GerParCor only included the metadata of the spellchecking:

1. The *good quality* category excludes both skipped and unknown words.

2. The *unknown good quality* category includes all words that are not skipped.

3. The three output percentages for correct, wrong, and unknown words are calculated based on all words that are not skipped (percentWords).

The integrated spellchecker in DUUI now stores and marks the spelling type for every token. If the spelling type is incorrect, it also saves the suggestions provided by *SymSpell*. It also provides additional metadata such as the counts of correct, wrong, unknown, and skipped words. In addition to the percentages of words (percentWords), it calculates the percentages of words including the skipped ones (percentWordsWithSkipped). The attribute *quality* includes in contrast to the other qualities the skipped words.

5. Provisioning

Based on the feedback after the release of GerParCor, it was observed that there is a technological challenge induced by UIMA. Thus, this update focuses on providing the corpus using simpler formats. The following new formats and access options are provided:

```
<type5:Sentence xmi:id="72" sofa="1413539" begin="131" end="133"/>         1
<type5:Token xmi:id="616159" sofa="1413539" begin="131" end="133"          2
   parent="616159" lemma="758336" pos="813101" morph="879135"
   order="0"/>
<type5:Lemma xmi:id="758326" sofa="1216419" begin="131" end="133"          3
   value="yr"/>
<pos:POS xmi:id="813091" sofa="1216419" begin="131" end="133"              4
   PosValue="XY" coarseValue="X"/>
 <morph:MorphologicalFeatures xmi:id="879125" sofa="1216419"               5
    begin="131" end="133" value=""/>
<dependency:ROOT xmi:id="1206812" sofa="1216419" begin="131" end="133"     6
   Governor="616149" Dependent="616149" DependencyType="--"
   flavor="basic"/>
<type:SuggestedAction xmi:id="1216803" sofa="1216419" begin="131"          7
   end="133" replacement="er" certainty="1.0"/>
<annotation2:AnomlySpelling xmi:id="1216809" sofa="1216419" begin="131"    8
   end="133" suggestions="1216803" category="Symspell"
   SpellingType="wrong" ModelName="Symspell"/>
<annotation2:AnomalySpellingMeta xmi:id="1413525" sofa="1216419"           9
   ModelName="Symspell" GoodQuality="0.6074789810515748"
   UnknownQuality="0.5655373831775701" Quality="0.4421408347794319"
   RightWords="4841" WrongWords="3128" UnknownWords="591"
   SkippedWords="2389" PercentRight="0.5655373831775701"
   PercentWrong="0.36542056074766355"
   PercentUnknown="0.06904205607476635"
   PercentRightWithoutSkipped="0.4421408347794319"
   PercentWrongWithoutSkipped="0.28568819070234724"
   PercentUnknownWithoutSkipped="0.05397753219472098"/>
<type12:Sentiment xmi:id="1471860" sofa="1413539" begin="0" end="61327"    10
   sentiment="0.007683863885839737" subjectivity="0.0"/>
<type12:Sentiment xmi:id="1413556" sofa="1413539" begin="0" end="60"       11
   sentiment="0.0" subjectivity="0.0"/>
```

Figure 3: Excerpt from an annotated XMI document from GerParCor. This exemplary XMI document contains all standoff annotations for a plenary protocol. The individual lines annotate different linguistic features, with the expansion being found starting from line 7 in particular. The sentence annotated in line 1 includes an annotated token (line 2) and other annotations from spaCy (1-6). In this example, the recognized token includes the string “yr”. *SymSpell* identified this token as incorrect and suggested a replacement “er” (7-8). Furthermore, after spell checking the token, associated metadata is annotated to be able to reuse it (9). A sentiment annotation for sentences is annotated in lines 10-11.

</> I A **lightweight web-based** interface[3] was implemented based on the migration of all parliamentary protocols into a UIMA database using (Abrami and Mehler, 2018), which allows users to a) browse the corpus in order to formulate queries using an API (</> III) and b) download individual parliamentary protocols in the provided export formats (</> II).

</> II Utilizing the GerParCor database - in combination with DUUI- **further export formats** can be provided flexibly and on runtime in addition to the existing UIMA XMI export. Since the plenary protocols are stored in UIMA, conversion into different file formats can be flexibly performed at any time, whereby CoNLL as well as Plain text are currently being provided. In addition, other export formats may also be provided, related to </> III and </> IV and the open source release of the project via GitHub.

</> III In addition to the different export formats (</> II), those interested can also use the provided **API in Java**. These API[4] (see Figure 4) can be used to perform selective queries on the corpus, so that within individual parliaments, as well as additionally for a period of time, the individual documents of the corpus can be

[3] https://gerparcor.texttechnologylab.org

[4] https://github.com/texttechnologylab/GerParCorAPI

Parliament	Periods	Sessions	Token	Sentences
Germany - National Level				
Bundestag	1949-07-09—-2023-10-19	3 784	253 011 771	16 145 907
Bundesrat	1949-07-09—-2023-07-07	1 034	32 770 581	2 542 619
Germany - Federal level				
Baden Württemberg	1952-03-25–2023-07-19	1 411	87 432 504	6 686 017
Bayern	1946-12-16–2023-07-20	2 435	121 107 176	9 498 137
Berlin	1947-10-30–2023-06-29	615	49 261 998	4 105 261
Brandenburg	1990-10-26–2023-02-23	472	35 023 783	2 672 329
Bremen	1933-02-01–2023-03-22	1 086	63 556 596	4 444 465
Hamburg	1997-10-08–2023-05-24	618	33 194 830	2 384 464
Hessen	1947-02-04–2023-07-19	1 906	116 590 626	9 378 830
Mecklenburg Vorpommern	1990-10-26–2023-03-21	806	57 832 474	3 999 065
Niedersachsen	1982-06-22–2023-06-23	1 149	84 630 836	6 653 024
Nordrhein Westfalen	1922-07-29–2023-06-16	2 104	119 825 597	9 129 715
Rheinland Pfalz	1909-02-03–2023-05-10	1 598	77 403 705	5 794 897
Saarland	1959-06-19–2023-06-21	880	49 083 469	3 269 933
Sachsen	1990-10-27–2023-07-06	728	55 445 597	4 220 679
Sachsen-Anhalt	1990-10-28–2023-06-30	650	48 472 896	3 838 504
Schleswig Holstein	1946-02-26–2022-04-28	1 840	95 883 840	7 467 113
Thüringen	1990-10-25–2023-09-30	862	54 918 174	3 322 782
Austria - National Level				
Nationalrat	1918-10-21–2023-09-29	3 749	237 633 328	16 897 525
Bundesrat	1920-12-01–2023-07-07	1 154	54 153 318	3 390 363
Austria - Federal Level				
Kärnten	1994-04-19–2023-04-13	389	28 403 753	1 805 111
Niederöstereich	1945-12-12–2023-07-06	764	32 005 485	2 198 781
Oberöstereich	1945-12-13–2023-05-11	569	24 683 177	1 367 965
Salzburg	1994-05-02–2023-02-01	216	9 387 717	574 290
Steiermark	1848-06-13–1968-06-17	1 797	28 021 868	1 656 910
Tirol	1865-11-23–2023-09-14	2 625	59 331 644	3 409 634
Voralberg	1822-04-29–2021-06-09	1 471	42 168 655	2 345 240
Wien	1998-01-23–2023-06-21	204	414 997	31 657
Switzerland - National Level				
Nationalrat	1999-06-12–2023-09-29	1 309	28 288 768	1 668 562
Liechtenstein - National Level				
Landtag	1997-03-13–2023-04-05	556	33 853 338	2 744 172

Table 2: Session periods of the individual regional and national parliaments in GerParCor which exist at present. The statistics given are from the date of publication of this paper. As we are working on a continuous expansion of GerParCor, the latest up-to-date statistics can be found at https://gerparcor.texttechnologylab.org.

downloaded and reused. An example for using the API, which can be directly integrated via Maven, is shown Figure 4. In addition, a REST-based API is provided which allows a machine-readable use of GerParCor.

</> IV Through the use of **</> I**, as well as the benefit of **</> II** in addition to the application of the functional scope of **</> III**, a **Reader** extension for DUUI is also a practical provision solution. Here the DUUI-Reader component provides the ability to process GerParCor data directly using DUUI without downloading it beforehand (e.g. as text or XMI files) to a local system. Simultaneously, a direct reuse by DUUI as well as its NLP capacities is then directly possible.

6. Corpus Statistics

There is significant growth in the volume of GerParCor, as shown in Table 2 for existing national and regional parliaments as well as in Table 3 for the historic. If plenary protocols, which are only available as digital copies, are to be converted into text files, they must be preprocessed using OCR. In sum, the corpus was increased by 28.35% (re-

```
// Initialization of the API.
GerParCorAPI pAPI = new GerParCorAPI();
// Initialize the factory to allow access to the corpus.
Factory pFactory = pApi.getFactory();
// Possible request to query all countries deposited in the corpus....
pFactory.listCountries().stream().forEach(sCountry->{
   // ... with subsequent filtering.
   QueryBuilder pQuery = new QueryBuilder();
   pQuery.withCountry(sCountry).withDevision("National")
   .withStartDate(pDate);

   Set<Protocol> pResult = pQuery.build();
   pResult.stream().forEach(p->{
      // Download a protocol
      File pFile = p.download(Format.XMI);
      // ...
   })
});

// In contrast, entire batch processes can be run to download the
    requested protocols in different contexts.
// Download all protocols from Germany as well as from Austria on
    national level.
QueryBuilder pQuery = new QueryBuilder();
pQuery.withCountry("Germany").withDevision("Regional");

// After the QueryBuilder is built, the desired protocols are written
    to the location of choice in the desired format.
pFactory.download(pQuery, Format.XTX, "/opt/corpus/");

// Alternatively, if protocols have already been downloaded to this
    location, avoiding overwriting will only add new protocols.
pFactory.download(pQuery, Format.XMI, "/opt/corpus/", false);
```

Figure 4: Example of GerParCor Java API usage.

garding sentences) compared to the previous version. In addition, we performed a sentiment analysis on sentence level, resulting in an average score of 0.10 in relation to the entire corpus, which allows better conclusions to be drawn at document or subcorpus level. Especially with old plenary protocols, i.e. from a period when only digital copies of moderate quality are available, it is important to calculate the OCR-quality.

Table 4 illustrates the output quality of all parliamentary documents extracted using Tesseract. Some readable OCR scans contained numerous errors due to the challenging Fraktur script. To address this issue, these scans were also processed using Tesseract for extraction. The Fraktur **Bundesrat (Austria)** has the worst quality score (69.15% – unknown good quality) and Baden Württemberg the best one (94.92% – good quality). The quality of Fraktur naturally suffers because *SymSpell* does not support Fraktur script. Nevertheless, our tests demonstrate that OCR is sufficiently accurate to support NLP based on new extended GerParCor.

7. Future Work

Generating and processing a corpus as large as GerParCor leads to a number of extensions which are useful extensions. These include both content-related and operational enhancements:

In order to expand the content of the corpus, it is advisable to add further parliamentary documents, so that besides the existing plenary protocols, printed matters, questions and committee protocols are also included. This additions would lead to even more growth and volume of GerParCor, which can be subsequently preprocessed using NLP methods. These further NLP preprocessing steps (e.g. topic analysis, semantic-role labeling) will be integrated into DUUI as NLP analysis methods in order to process GerParCor- as well as other corpora, of course - in a uniformed

Parliament	Periods	Sessions	Token	Sentences
	Germany - National Level			
Reichstag (North German Union / Zollparlamente)	1867-02-25–1895-05-24	1 970	76 593 232	4 430 065
Reichstag (German Empire)	1895-03-12–-1918-10-26	2 183	60 102 498	3 096 673
Weimar Republic	1919-02-06–-1932-09-12	1 331	44 408 757	2 888 948
Third Reich	1933-21-03–-1942-04-24	9	186 955	11 998
	Germany - Federal level			
Alter Landtag Württemberg	1797-04-24–1799-01-30	19 (📦)	1 173 546	68 186
Landtag Württemberg	1820-01-18–1933-10-16	381	159 894 169	9 172 003
Landtag Württemberg-Baden	1946-12-10–1952-05-30	11 (📦)	7 293 642	517 172
Landtag Württemberg-Hohenzollern	1947-06-03–1952-05-30	5 (📦)	2 613 376	179 772
Ständeversammlung Württemberg	1815-03-15–1891-09-25	49	2 232 113	147 158
VGL Baden-Württemberg	1952-03-25–1953-11-11	3 (📦)	2 516 838	189 249
VGL Württemberg	1849-12-01–1920-05-21	6 (📦)	3 765 383	228 212
VGL Württemberg-Baden	1946-01-16–1946-06-19	1 (📦)	304 177	16 596
VGL Württemberg-Hohenzollern	1946-11-22–1947-05-09	1 (📦)	299 201	18 784

Table 3: Overview of historical parliamentary protocols in GerParCor, for parliaments which no longer exist. VGL is a acronym for "Verfassungsgebende Landesversammlung", which meens "Constitutional State Assembly". If there is 📦 in the column with the number of sessions, this means that the sessions are only available in collections and not individually.

Parliament	**Period**	**good quality**	**unknown good quality**	**unknown words %**	**right words %**	**wrong words %**
Baden Württemberg	1952-03-25–1984-05-08	94.92%	89.17%	6.06%	89.17%	4.77%
Baden Württemberg	1797-04-24–1996-02-08	81.49%	76.72%	5.85%	76.72%	17.43%
Bundesrat (Austria)	1920-12-01–1934-04-30	75.21%	69.15%	8.06%	69.15%	22.8%
Niederöstereich	1945-12-12–1964-02-12	92.80%	86.98%	6.28%	86.98%	6.75%
Oberöstereich	1945-12-13–1955-10-27	93.32%	87.21%	6.54%	87.21%	6.24%
Oberöstereich	1955-11-19–1991-07-03	74.39%	68.99%	7.27%	68.99%	23.75%
Steiermark	1848-06-13–1938-01-17	75.21%	69.96%	6.98%	69.96%	23.06%
Tirol	1865-11-23–1969-11-25	75.52%	70.31%	6.89%	70.31%	22.79%

Table 4: Testing OCR quality based on Tesseract. Bold face refers to Fraktur.

approach. Moreover employing the Fraktur script of GerParCor to create a spellchecking tool enhances the correction of words in OCR Fraktur script scans, thereby improving their accuracy, for further processes. In addition, this will provide the basis for further analysis as well as upcoming training tasks. As an addition to the existing Java and REST API, it would also be reasonable - although this is only of secondary benefit due to the use of DUUI- to implement a Python API as well.

However, in the intermediate perspective, the most challenging issue is to address the heterogeneity problem of the individual services for parliamentary documentation. These should be an important addition not only for GerParCor but also in general, including Open Data efforts in each country. Since there are already ethablized methods (e.g. CMDI (de Vries et al., 2021)) as well as robust harvesting methods (e.g. OAI-PMH (Kolosov, 2022)), a reuse is certainly appropriate. Although this would not have to be done for past plenary documents, but as of a future point in time – for everything else there is GerParCor.

8. Conclusion

With this update of GerParCor, we were able to significantly expand the largest German-language parliamentary corpus to date, not only by adding new plenary protocols of the national and regional parliaments already covered, but also by adding entirely new parliaments. Through this expansion in the historical area, we can now access parliamentary debates back to the year 1797 for the German-speaking countries as well as include all available protocols of the regional parliaments for Austria. Furthermore, the introduction of a wide range of provisioning possibilities supports a more flexible and active reuse of GerParCor. In addition to a new Java API and lightweight web interface, a reader is also available for further use with DUUI. At the same time, new DUUI components have been developed in terms of Docker images, which can also be obtained and used directly on different applications. Obligatory, but also provisioning, the source code of GerParCor as well as for the Java API is available via GitHub under AGPL license for reuse and extension. The provision of the parliamentary protocols is intended to trigger a variety of analyses and investigations based on GerParCor.

Ethical Consideration

This work has been developed considering ethical aspects. With our contribution, we intend to make a contribution with regard to providing German-language corpora for parliamentary documents and strive to map the information as completely as possible. Although we only refer to existing and publicly documented protocols, individuals may be exposed to content that is hurtful, condescending, and contemptuous through the use of the Corpus. Moreover, our technology is not limited to this corpus and preprocessing - also by means of DUUI- can also be applied to text corpora which may be prohibited from processing. The authors are aware of this situation, but we also respect free research.

9. Bibliographical References

Giuseppe Abrami, Mevlüt Bagci, Leon Hammerla, and Alexander Mehler. 2022. German parliamentary corpus (gerparcor). In *Proceedings of the Language Resources and Evaluation Conference*, pages 1900–1906, Marseille, France. European Language Resources Association.

Giuseppe Abrami and Alexander Mehler. 2018. A uima database interface for managing nlp-related text annotations. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference, May 7 - 12*, LREC 2018, Miyazaki, Japan.

Adrien Barbaresi. 2018. A corpus of german political speeches from the 21st century. In *International Conference on Language Resources and Evaluation*.

Francesco Barbieri, Luis Espinosa Anke, and Jose Camacho-Collados. 2022. XLM-T: Multilingual language models in Twitter for sentiment analysis and beyond. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 258–266, Marseille, France. European Language Resources Association.

Eduoard Belval. 2017. symspellpy. https://github.com/Belval/pdf2image. Accessed: 2022-01-17.

Andreas Blaette and Christoph Leonhardt. 2023. Germaparl corpus of plenary protocols.

Tobias Bornheim, Niklas Grieger, Patrick Gustav Blaneck, and Stephan Bialonski. 2023. Speaker attribution in german parliamentary debates with qlora-adapted large language models.

G. Bradski. 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Jerry de Vries, Vyacheslav Tykhonov, Andrea Scharnhorst, Eko Indarto, Femmy Admiraal, and Mike Priddy. 2021. Flexible metadata schemes for research data repositories.the common framework in dataverse and the CMDI use case. In *Selected Papers from the CLARIN Annual Conference 2021, virtual event, September 27-29, 2021*, volume 189 of *Linköping Electronic Conference Proceedings*, pages 168–180. Linköping University Electronic Press.

Tomaž Erjavec, Maciej Ogrodniczuk, Petya Osenova, Nikola Ljubešić, Kiril Simov, Andrej Pančur, Michał Rudolf, Matyáš Kopp, Starkaður Barkarson, Steinþór Steingrímsson, Çağrı Çöltekin, Jesse de Does, Katrien Depuydt, Tommaso Agnoloni, Giulia Venturi, María Calzada Pérez, Luciana D. de Macedo, Costanza Navarretta, Giancarlo Luxardo, Matthew Coole, Paul Rayson, Vaidas Morkevičius, Tomas Krilavičius, Roberts Darundefinedis, Orsolya Ring, Ruben van Heusden, Maarten Marx, and Darja Fišer. 2022. The parlamint corpora of parliamentary proceedings. *Lang. Resour. Eval.*, 57(1):415–448.

David Ferrucci, Adam Lally, Karin Verspoor, and Eric Nyberg. 2009. Unstructured Information

Management Architecture (UIMA) Version 1.0. OASIS Standard.

Wahed Hemati, Tolga Uslu, and Alexander Mehler. 2016. Textimager: a distributed uima-based system for nlp. In *Proceedings of the COLING 2016 System Demonstrations*. Federated Conference on Computer Science and Information Systems.

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

Kirill A. Kolosov. 2022. Integrating with open archive via oai-pmh provider. *The letter and digit: The libraries on the way to digitalization*.

Alexander Leonhardt, Giuseppe Abrami, Daniel Baumartz, and Alexander Mehler. 2023. Unlocking the heterogeneous landscape of big data NLP with DUUI. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 385–399, Singapore. Association for Computational Linguistics.

Dean Malmgren. 2014. textract. `https://textract.readthedocs.io/en/stable`. Accessed: 2023-10-19.

mammothb. 2018. symspellpy. https://github.com/mammothb/symspellpy. Accessed: 2022-01-17.

Christian Rauh and Jan Schwalbach. 2020. The ParlSpeech V2 data set: Full-text corpora of 6.3 million parliamentary speeches in the key legislative chambers of nine representative democracies.

Colin Sippl, Manuel Burghardt, Christian Wolff, and Bettina Mielke. 2016. Korpusbasierte analyse österreichischer parlamentsreden.

Kristina Toutanvoa and Christopher D. Manning. 2000. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. In *2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 63–70, Hong Kong, China. Association for Computational Linguistics.

Naomi Truan and Laurent Romary. 2021. Building, encoding, and annotating a corpus of parliamentary debates in tei xml: A cross-linguistic account. *Journal of the Text Encoding Initiative*.

Karsten Tymann, Matthias Lutz, Patrick Palsbröker, and Carsten Gips. 2019. Gervader - A german adaptation of the VADER sentiment analysis tool for social media texts. In *Proceedings of the Conference on "Lernen, Wissen, Daten, Analysen", Berlin, Germany, September 30 - October 2, 2019*, volume 2454 of *CEUR Workshop Proceedings*, pages 178–189. CEUR-WS.org.

Tanja Wissik and Hannes Pirker. 2018. Parlat beta corpus of austrian parliamentary records.