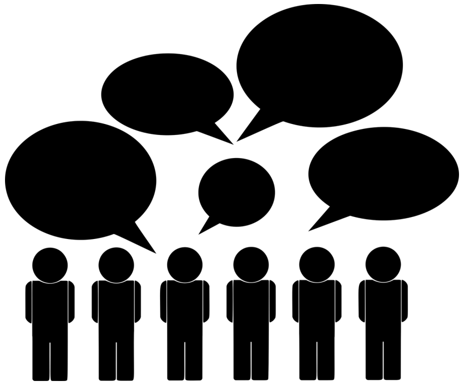
Proceedings of the GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates (SpkAtt-2023)



Ines Rehbein ♠, Fynn Petersen-Frey ♣, Annelen Brunner ♢,
Josef Ruppenhofer ♡, Chris Biemann ♣, Simone Paolo Ponzetto ♠
University of Mannheim ♠, FernUniversität Hagen ♡,
Leibniz Institute for the German Language ♢, University of Hamburg ♣

Sep 18, 2023

*Proceedings of the GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates (SpkAtt-2023), ,
Ingolstadt, Germany, Sep 18, 2023*

*Proceedings of the GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates (SpkAtt-2023), ,
Ingolstadt, Germany, Sep 18, 2023*

Workshop Program

September 18, 2023

15:00 – 17:00 GermEval Shared Task on Speaker Attribution

17:00 – 17:05 Closing & Good-Bye

Overview of the GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates

Ines Rehbein DWS Mannheim U rehbein@uni-mannheim.de

Fynn Petersen-Frey HCDS Hamburg U fynn.petersen-frey@uni-hamburg.de

Annelen Brunner Lexik Leibniz-IDS brunner@ids-mannheim.de

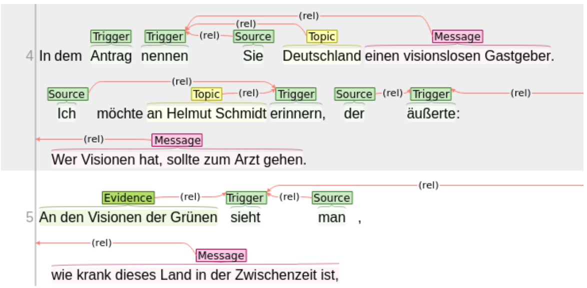
Josef Ruppenhofer CATALPA Hagen U josef.ruppenhofer@fernuni-hagen.de

Chris Biemann LT Hamburg U chris.biemann@uni-hamburg.de

Simone Paolo Ponzetto DWS Mannheim U ponzetto@uni-mannheim.de

Abstract

This paper gives an overview of the GermEval 2023 Shared task on Speaker Attribution in Newswire and Parliamentary Debates (SpkAtt2023) and describes the data, annotation guidelines and results of the evaluation campaign. The task targets the identification of speech events in text and their attribution of the respective speakers, including the detection of other roles that might be expressed, such as the addressee or the topic of the speech event. The shared task includes two subtasks, (i) the identification of speech, thought and writing in parliamentary debates and (ii) in newswire text. Being able to identify *who* says *what* to *whom* is crucial for in-depth analyses and enables researchers to extract more meaningful information from unstructured text.



In dem Antrag nennen Sie Deutschland einen visionslosen Gastgeber.

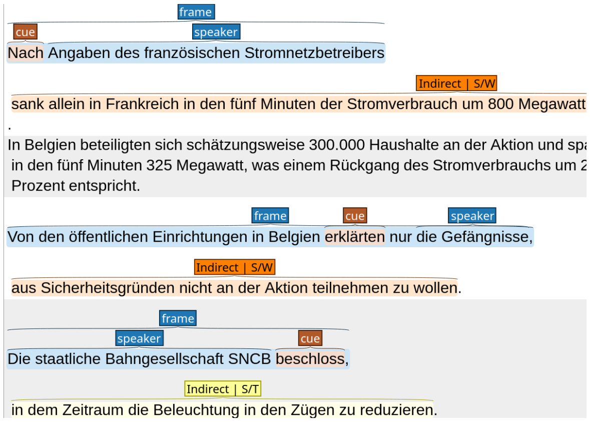
Ich möchte an Helmut Schmidt erinnern, der äußerte:

Wer Visionen hat, sollte zum Arzt gehen.

An den Visionen der Grünen sieht man ,

wie krank dieses Land in der Zwischenzeit ist,

Figure 1: Example for speaker attribution in parliamentary debates (Task 1).



Nach Angaben des französischen Stromnetzbetreibers

sank allein in Frankreich in den fünf Minuten der Stromverbrauch um 800 Megawatt

.

In Belgien beteiligten sich schätzungsweise 300.000 Haushalte an der Aktion und spa
in den fünf Minuten 325 Megawatt, was einem Rückgang des Stromverbrauchs um 2
Prozent entspricht.

Von den öffentlichen Einrichtungen in Belgien erklärten nur die Gefängnisse,

aus Sicherheitsgründen nicht an der Aktion teilnehmen zu wollen.

Die staatliche Bahngesellschaft SNCB beschloss,

in dem Zeitraum die Beleuchtung in den Zügen zu reduzieren.

Figure 2: Example for speaker attribution in news articles (Task 2).

1 Introduction

Identifying who says what to whom is an essential prerequisite for analysing human communication. The complexity of the task, however, is often underestimated by assuming that the words produced by the speaker only reflect his or her own point of view. Figure 1 shows an excerpt from a parliamentary debate of the German Bundestag, illustrating how speakers frequently switch perspectives, at times presenting their own views and sometimes reporting and citing the views of others. Thus, it is crucial to identify the correct source for each speech event when analysing text. Furthermore, studying how speakers construct their own arguments relative to the views of other speakers, either to back up their own claim or to attack the others' perspective, is an intruiging research question in itself.

In order to investigate these questions, we need annotated resources that allow us to train models that learn to predict speech events in unstructured text, together with their respective speakers, messages and addressees. This overview paper presents two new resources for speaker attribution in German text, based on parliamentary debates from the German Bundestag and on newswire text. We first review previous work on quote detection and speaker attribution before we describe our data and the annotation process. Then we provide a description of the shared task settings and report baseline results for each of the two new resources. Finally, we present the results of the shared task, with an evaluation of the system output for the participating systems.

Cue/Role name	description	example
CUE	the cue that triggers the STW event	Merkel spoke to the people.
SOURCE	Source of the STW event	Merkel spoke to the people.
MEDIUM	Medium of the STW event	The Basic Law reads ...
MESSAGE	Message / content of the STW event	She said that she would resign.
TOPIC	Topic of the STW event	Merkel addressed the theme of taxation.
EVIDENCE	Evidence for the message	The survey shows that ...
ADDRESSEE	Addressee of the STW event	Merkel spoke to the people.
PARTICLE	Separated verb prefix or	Merkel schlug vor (proposed) ...
(PTC)	obligatory particle	Merkel CUE sich vor (imagines herself) ...

Table 1: Overview over our classification scheme for annotating events of **S**peech, **T**hought and **W**riting (STW).

2 Related Work

2.1 Work on speaker attribution

Much recent work has been devoted to quote detection, mostly with the goal of extracting information from newswire text (Pouliquen et al., 2007; Krestel et al., 2008; Pareti et al., 2013; Pareti, 2015; Scheible et al., 2016). Other related work comes from the field of opinion mining and has targeted the identification of opinion holders (speakers) and the targets of the opinions (Choi et al., 2005; Wiegand and Klakow, 2012; Johansson and Moschitti, 2013).

Many studies have addressed speaker attribution in novels and other literary works, in the context of computational literary studies. Elson and McKeown (2010) were among the first to propose a supervised machine learning model for quote attribution in literary text. He et al. (2013) extended their supervised approach by including contextual knowledge from unsupervised actor-topic models. Almeida et al. (2014) and Fertmann (2016) combined the task of speaker identification with coreference resolution. Grishina and Stede (2017) test the projection of coreference annotations, a task related to speaker attribution, using multiple source languages. Muzny et al. (2017) improved on previous work on quote and speaker attribution by providing a cleaned-up dataset, the QuoteLi3 corpus, which includes more annotations than the previous datasets. They also present a two-step deterministic sieve model for speaker attribution on the entity level and report a high precision for their approach.[1] Papay and Padó (2020) annotate direct and indirect quotations in 19th century English literature while Kim and Klinger (2018) extend the speaker attribution task to capture emotion trigger phrases and the experiencers, targets and causes of the emotion.

While many studies have addressed the task of quote detection or speaker attribution in English text from the literary domain or in news articles, less work has been done for other languages and genres. Brunner (2015), Krug et al. (2018), Brunner et al. (2019) and Brunner et al. (2020) have focused on German literary text and created several resources. The DROC corpus (Krug et al., 2018) includes around 2,000 manually annotated quotes and annotations for speakers and their mentions in 90 fragments from German literary prose and the RedeWiedergabe corpus substantially extends this work by presenting a German-language historical corpus with detailed annotations for speech, thought and writing (Brunner et al., 2020). Dönicke et al. (2022) address a task related to speaker attribution, i.e., identifying whether a certain text passage is written from the perspective of the narrator of the novel or from the author's point of view, or whether it reflects the view of a character in the novel. Interestingly, they show that including annotator bias in the model can improve results.

Less work has been done for other domains. A noteworthy exception is Ruppenhofer et al. (2010) who present preliminary work on speaker attribution in text from the political domain, using German cabinet protocols. As the focus of SpkAtt2023 Task 1 is also on analysing the language of political debates, we extended the work of Ruppenhofer et al. (2010) and created a new, manually annotated resource for speaker attribution with around 13,000 clauses and more than 200,000 tokens. Our second research focus is on analysing who says what to whom according to German news media (SpkAtt2023 Task 2). For this, we created a new, manually annotated dataset for speaker attribution in German news articles with almost 250,000 tokens. For both resources, our annotation follows (Brunner, 2015; Brunner et al., 2020) and considers

[1] When optimised for precision, the system obtains a score >95% on the development set from *Pride and Prejudice*.

Cue/Role	freq.	avg. len
CUE	7,706	1.1
SOURCE	4,663	1.7
MESSAGE	4,578	9.7
TOPIC	1,188	5.4
ADDRESSEE	717	3.2
PARTICLE	561	1.0
MEDIUM	321	3.2
EVIDENCE	151	4.3

Table 2: Statistics for the Task 1 dataset (GePaDe).

	overlap		*binary*
Sample	**Cue**	**Roles**	**Roles**
Sample 1	69.07	64.53	67.88
Sample 2	81.19	67.04	72.60
Sample 3	81.95	72.11	76.90
Sample 4	82.84	73.81	77.63

Table 3: Pair-wise percentage agreement between the annotators on the four samples from GePaDe (Task 1) (*overlap:* proportional token overlap between A1 and A2; *binary:* at least one token in the cue/role span has been identified and assigned the same label).

not only speech events, but also thought and writing. We describe the creation of these resources in the next section.

3 Data and Annotation

3.1 Task 1: Speaker attribution in German parliamentary debates

We present a new dataset for speaker attribution in data from the political domain, specifically, parliamentary debates from the German Bundestag. Our dataset includes manually annotated cues that trigger events of speech, writing and thought.[2] In addition, we annotate the arguments of the trigger, including the SOURCE, ADDRESSEE, MESSAGE, MEDIUM, TOPIC and EVIDENCE for the speech event. Table 1 shows examples for the different categories in our schema.[3] We now describe our data, annotation setup and annotation procedure.

Data The data for Task 1 includes debates from the German Bundestag, retrieved from Deutscher Bundestag – Open Data.[4] The data set includes 265 speeches from the German Bundestag, mostly from the 19th legislative term (2017-2021), given by 195 different speakers from 6 parties (CDU/CSU: 76, SPD: 57, AfD: 39, FDP: 33, Linke: 29, Grüne: 26, non-attached: 4). The total size of the data is >200,000 tokens. For more detailed information on the data, sampling and annotation process, please refer to the datasheet.[5]

Annotation process The data was annotated by four student assistants from different fields in the humanities. The annotators received extensive training. During the annotation phase, weekly meetings were held where we discussed open questions and problematic cases.

To ease the detection of speech events, we started with a list of cue words extracted from the RedeWiedergabe Corpus (Brunner et al., 2020). We marked all lemma forms from the list in our data and instructed the annotators a) to verify whether this instance is a Speech, Thought and Writing (henceforth: STW) event and, b) if true, to identify all of its arguments realised in the utterance. To increase recall, we asked the annotators to add new cue words to the list that were then included in the annotation. Table 2 shows the number of annotated cues and their roles in our corpus. Overall, we annotated more than 7,700 events of speech, thought or writing in the data.

Inter-annotator agreement We split the data into four samples that reflect the order of annotation. Table 3 shows the average percentage agreement of two coders for cue words and roles as the proportional token *overlap* between the annotated cues or roles. To augment this view, we also report a more lenient *binary* score which considers an annotation as correct if at least one token in the annotations overlaps and has been assigned the same label.[6] We can clearly see that inter-annotator agreement constantly improves with more training even after the third round of annotation.

Disagreements between the annotators Most questions during annotation concerned the class

[2] In the reminder of the paper, we use the term "speech event" to refer to events of speech, thought or writing.

[3] The annotation guidelines are available at https://github.com/umanlp/SpkAtt-2023.

[4] https://www.bundestag.de/services/opendata.

[5] The datasheet is available from our github repository: https://github.com/umanlp/SpkAtt-2023/blob/master/doc/SpkAtt-Debates-Datasheet.pdf.

[6] For more details on the scoring method, see (Marasovic and Frank, 2018).

of Thought events. Our guidelines follow Brunner et al. (2020) and define Thought as "silent or inner speech which can be reproduced in the same way as verbalized speech". Brunner et al. (2020) conceptualise Thought as "a conscious, analytical, cognitive process" and exclude descriptions of emotional and mood states or passages that are told from a strongly personal perspective. This definition, however, is hard to operationalise and there were many borderline cases that required discussion. We used our weekly meetings to decide which new cue words we would like to include. For more details, please refer to the annotation guidelines.

At the beginning of the annotation process, some annotators were eager to identify new cue words for thought events while others had a more conservative approach, considering only cues from our list. This is reflected in the high disagreement for sample 1. Sometimes new cues were included after one coder had already completed a document, ignoring those cues, while the second coder included the new cues in the annotations. The confusion matrix (Appendix, Table 11) shows that this is in fact the major source of disagreements: instances that were annotated by one annotator but not by the second coder (label None).

Other disagreements concern the distinction between Message and Topic (Example 3.1) and between Medium and Evidence (Example 3.2).

When distinguishing between Topic and Message, the annotators sometimes struggled to decide whether the speaker simply mentioned a certain topic or whether she also tried to convey a message. For instance, Example 3.1 may either be taken to mean that the addressee ("Sie" , 2Sg.formal) spoke about a democratic imposition (Topic) or that they said that something constituted a democratic imposition (Message).[7] Similarly, the distinction between Medium and Evidence was another case that was difficult for the annotators. Consider Example 3.2 where it is not clear whether the bold-faced text should be considered as the medium that transported the message or whether it should be interpretated as Evidence. More details on the distinction between those labels can be found in the annotation guidelines.

Ex. 3.1 (Topic vs. Message)

Sie haben von einer „demokratischen Zumutung“ gesprochen.
You have spoken **of a "democratic imposition"**.

Ex. 3.2 (Medium vs. Evidence)

[...] die weltweite Stimmung mahnt uns, Erkämpftes zu erhalten [...]
[...] the global mood *urges us to preserve what we have fought for [...]*

	freq.	**avg. len**
sentence	13,186	18.84
Message	4,182	16.69
Cue	2,929	1.57
Addressee	337	2.72
Frame	3,038	8.95
Source	3,908	3.53

Table 4: Statistics for the Task 2 dataset (news).

3.2 Task 2: Speaker Attribution in German news articles

We present a new creative-commons-licensed dataset for speaker attribution in German news articles. The dataset consists of manually annotated articles from the German WIKINEWS website.[8] In total, these annotated articles contain almost 250,000 tokens. We manually annotated and curated Messages in different forms of speech such as Direct, Indirect, Free Indirect, Indirect/Free Indirect, Reported together with the corresponding Frame, Source, Cue and Addressee. Table 4 reports the number and the average length of Message and the four roles used in Task 2. Examples for these roles can be found in Table 1. Table 5 shows the number and average length of the Speech/Thought/Writing representation (STWR) and the form of speech for our Message annotations. In the following subsections, we describe the raw source data, its pre-processing, the annotation process, the inter-annotator agreement and the handling of disagreements between annotators.

3.2.1 Source data

The data originates from news articles published on the German WIKINEWS website. We used the XML dump[9] available through the Wikimedia foundation. Our dataset is based on the dump from

[7] Based on the quotation signs used we think the latter interpretation is more likely to be correct but it's a subtle judgment.

[8] URL: https://de.wikinews.org

[9] URL: https://dumps.wikimedia.org/dewikinews/

	freq.	avg. len
DIRECT	873	17.54
INDIRECT	2250	14.71
FREE INDIRECT	171	20.43
INDIRECT/FREE INDIRECT	434	22.33
REPORTED	454	18.01
SPEECH	1906	16.75
WRITING	572	19.13
THOUGHT	2	10.5
SPEECH/THOUGHT (ST)	322	14.95
SPEECH/WRITING (SW)	1362	16.0
WRITING/THOUGHT (WT)	0	-

Table 5: MESSAGE statistics for the Task 2 dataset.

April 2022 that consists of 13,001 published articles. From these published articles, we sampled 1000 articles to annotate. These articles range from December 2004 to March 2022.

3.2.2 Data pre-processing

Since the articles are stored in MediaWiki markup with custom macros for the German WIKINEWS, we wrote a program to automatically convert this markup into plain text. The conversion is a recursive procedure in order to support the nested macros present in the markup. Using this approach, we stripped all markup like formatting (e.g. bold, italic), semantic information (e.g. links to entities on Wikipedia) and non-textual content (e.g. pictures, tables) from the documents. Further, we removed any text not belonging to the main text body such as publication metadata, comments, links to related articles or sources. The resulting plain text was tokenized and split into sentences using spaCy (Honnibal et al., 2020). Finally, the tokenized text was exported in a format compatible with our annotation software.

3.2.3 Annotation process

The annotation was carried out by three annotators with a background in German studies or Linguistics and an additional supervisor. The annotators were selected after performing a trial annotation on a handful of articles. The annotation team received training during a preliminary annotation before the actual annotation begun. Further, we held weekly meetings during the main annotation to discuss open questions and uncertain cases, thereby providing ongoing training to all annotators.

As outlined in Section 2, the annotation scheme is based on the Redewiedergabe project (Brunner et al., 2020). In an initial preliminary annotation, we tested the suitability of the annotation scheme in the news domain. We iteratively tested which attributes of the schema are necessary and which additional options we needed. Finally, we settled on the medium (referred to as STWR in the dataset) and type attribute for a MESSAGE and FRAME, CUE, SOURCE and ADDRESSEE as the other annotation parts (roles). STWR can either be SPEECH, THOUGHT, WRITING or one of the combinations SPEECH/THOUGHT, SPEECH/WRITING, WRITING/THOUGHT for cases where it is not possible to confidently decide on a single value from the text. The types of speech are taken from the Redewiedergabe project: DIRECT, INDIRECT, FREE INDIRECT, REPORTED and INDIRECT/FREE INDIRECT. For more details refer to the annotation guidelines (see supplementary materials).

For the annotation, we used the annotation software INCEpTION (Klie et al., 2018). The different parts are modeled as span annotations with relations between them to indicate e.g. which SOURCE belongs to which MESSAGE.

Sample	Form	STWR	Roles
Sample 1	0.56	0.37	0.61
Sample 2	0.76	0.51	0.75
Sample 3	0.77	0.40	0.76
Sample 4	0.77	0.68	0.76
Sample 5	0.86	0.51	0.83
Sample 6	0.78	0.61	0.78

Table 6: Krippendorff's Alpha agreement between the annotators on the six samples from Task 2

3.2.4 Inter-annotator agreement

We used Krippendorff's Alpha to compute the agreement between two annotators per sample. The measure includes both the quality of the span annotation offsets (overlap) as well as their labels, but does not include the relations between the span annotations. However, the relations were typically made identically given the same annotation spans and labels. Moreover, for different annotation spans, there is no sensible way to compute an inter-annotator agreement on the relations.

Table 6 shows the inter-annotator agreement values for the six samples into which we divided the 1000 annotated documents. The inter-annotator agreement values increased strongly after the first sample, slightly increasing with additional experience and training over the course of the remaining samples. As such, the first sample required significant curation effort and discussion that ultimately led to improved skills of our annotators.

3.2.5 Disagreements between annotators

During the annotation phase we held weekly meetings to discuss general questions concerning how we would best annotate specific phenomena within our annotation scheme. After two annotators had finished annotating the documents, we employed curation by a third person to resolve differences between the annotations. In situations where the curator was not certain who (or if any) of the two annotators had annotated the sentences in question correctly, we discussed the issue in detail to resolve the disagreement, thereby potentially defining our annotation guidelines more precisely.

One of the most frequent reasons of disagreement during the early phases of the annotation was the difficulty of choosing the correct STWR, usually the choice being between writing or speech. After many discussions, we concluded that it is sometimes impossible to decide from the text alone whether an utterance was produced in spoken or written form. As such, we modified our annotation scheme by adding three new labels to STWR (see Section 3.2.3).

4 Task Description

The SpkAtt2023 shared task included two tasks: (i) speaker attribution for parliamentary debates from the German Bundestag and (ii) speaker attribution in German newswire. The teams could participate either in both or just in one of the two tasks.

The terms of the shared task required that any data or models used outside of those that are provided should be publicly accessible or be made public by April 1, 2023 (release of the training data). Each team could submit multiple submissions, however, the last submission uploaded by the team was considered to be the official entry to the competition.

4.1 Task 1: Parliamentary debates

The goal of Task 1 was the identification of speakers in political debates and newswire, and the attribution of speech events to their respective speakers.

For this task, participants were asked to build a system that can identify all cue words that trigger a speech event and, for each speech event, all roles associated with this event (i.e., Source, Addressee, Message, Topic, Medium, Evidence). The task setup is thus similar to Semantic Role Labelling.

For Task 1, the participants could take part in the following subtasks:

- **Subtask 1 (full task):** Participants were asked to predict the cue words that trigger a speech event, together with the associated roles and their respective labels.
- **Subtask 2 (role labelling):** For this subtask, the gold cue words were given and the task consisted in identifying the spans for all associated roles expressed in the text, together with their respective labels.

A detailed description of the data format and the annotations can be found in the Task 1 GitHub repository: `https://github.com/umanlp/SpkAtt-2023` (see README and annotation guidelines). The trial and training data were made available from the same GitHub page.

4.2 Task 2: News articles

For this task, participants had to develop a system that identifies statements (MESSAGE), i.e. instances of speech (DIRECT, INDIRECT, FREE INDIRECT, INDIRECT/FREE INDIRECT, REPORTED) and the corresponding roles with it (FRAME, SOURCE, ADDRESSEE, CUE). Further, the system should identify the speech form and relevant medium (SPEECH, THOUGHT, WRITING) accordingly.

The participants could take part in the following task settings:

- **Subtask 1 (full task)**: Predict all parts of a statement, associate them, and label the form of speech and medium
- **Subtask 2 (simplified)**: Predict only the SOURCE (i.e. speaker) and MESSAGE (quotation) of top-level (i.e. not nested) annotations, then link both together. The annotation data contains a boolean flag to select only relevant annotations (`"IsNested": false`)

The technical data format description and some additional details are provided in the Task 2 GitHub repository at `https://github.com/uhh-lt/news-speaker-attribution-2023` (see the README file). This website is the place where the trial, training, development and blind test data were published.

5 Evaluation

We now present the experimental setup and report baseline results for both tasks.

5.1 Baseline system (Task 1 – GePaDe)

In order to automatically predict cue words for speech events and their roles, we split our data into training, dev and test sets with 9,298/927/3,067 sentences.[10] This amounts to 178/18/72 different speeches in each set, with 5,536 (train), 515 (dev) and 3,646 (test) annotated STW events.

For our baseline, we use two heuristic approaches. To predict the cue words, we extract all wordforms for cues from the training data. To reduce noise, we do not consider multiword triggers and also remove prepositions from the set of cue words. Then we search the test data for wordforms that match a cue word from our list and, if we find one, we insert a speech event for this cue.

To predict the roles, we use a dependency-based syntactic heuristic and assign all subjects of verbal cue words the label SOURCE and all direct objects of verbal cue words the label MESSAGE. For nominal cue words, we assign the label SOURCE to possessive pronouns (Ihren eigenen Antrag; *engl.: her own proposal*) and genitive NPs that bear the dependency label AG.

5.2 Evaluation metrics

The evaluation of system performance uses the familiar Precision, Recall and F1-metrics. Both cue and role labels can cover more than one token and therefore are represented as sets of (possibly discontinuous) tokens. The annotation scheme assumes that a given set of tokens can bear at most one cue annotation, that is, it can evoke at most one instance of speech, throught or writing. For roles this is not true: a set of tokens could bear multiple role labels, usually in relation to different cues.

According to our definition of the task, roles are dependent on cues and so system roles can match gold roles only if they are related to the same cue. In line with this, the evaluation first checks how system cues and gold cues align. In doing so the scorer matches at most one system cue to a at most one gold cue and the same in the other direction. System cues that cannot be aligned to gold cues produce false positives, including for their associated roles. In symmetric fashion, gold cues that cannot be aligned to a system cue result in false negatives.

For both cues and roles, alignment requires nonzero overlap with the tokens covered by a label of the same type on the other side. Each component token of aligned labels is counted as a true or false positive, or as a false negative. This means that longer spans contribute more to the overall score than shorter labels. In situations where a multi-token cue on one side overlaps with two or more separate cues on the other side, the scorer scores all possible alignments and chooses the one that maximizes the joint F1-score for cues and roles.

5.3 Baseline system (Task 2)

We developed Quotes in Text (QUiTE) – a rule-based system to extract direct and indirect quotations with the speaker from text. The system follows ideas of an older system presented by Bögel and Gertz (2015). QUiTE uses rules and word lists on top of neural components for dependency parsing and named-entity recognition. DIRECT speech is identified by regular expressions looking for quotation marks. The SOURCE of the quotation (i.e. the speaker) is searched in the proximity, preferring candidates in the same sentence but outside of the quotation span. INDIRECT speech is identified through the grammatical structure of a sentence (using dependency parsing) and the main or auxiliary verb being a cue word that is looked up in a word list. The word list contains utterance verbs (verba dicendi) that can be used to indicate (in)direct speech. In addition, the system finds sentences in subjunctive mood that occur directly before or after a sentence containing quotation and source. These sentences are typically marked as INDIRECT/FREE INDIRECT in the dataset. Lastly, the system combines DIRECT and INDIRECT speech, enriching the information of identical quotations.

5.4 Evaluation metrics (Task 2)

Task 2 is evaluated similarly to Task 1 using the the usual Precision, Recall and F1-metrics on token overlap of possibly discontinuous spans (sets of tokens). However, the roles are optional but always depend on a MESSAGE. Thus, predicted roles can only match gold roles if they belong to a matched MESSAGE. A span representing a role can be related to multiple MESSAGE spans, i.e. the same SOURCE can utter multiple MESSAGES. Roles or MESSAGE spans can be nested within an-

[10] We use spaCy for sentence segmentation which results in segments on the clause level, with an average size of around 16 tokens/clause.

	Cues			**Roles**			**Joint**		
Team	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
baseline	57.34	82.96	67.81	67.02	32.00	43.32	64.33	37.73	47.56
aehrm2	89.70	88.87	89.28	77.64	87.06	82.08	78.85	87.26	82.84
nesasio	88.92	88.92	88.92	78.69	82.15	80.38	79.80	82.91	81.33
moiddes	67.48	66.08	66.77	56.67	84.30	67.78	57.51	82.25	67.69

Table 7: Evaluation results for Task1, subtask 1 (cues & roles).

Team	**Prec**	**Rec**	**F1**
baseline	89.08	33.66	48.86
aehrm2	91.12	90.23	90.67
nesasio	90.96	87.32	89.10
moiddes	53.49	85.35	65.76

Table 8: Evaluation results for Task1, subtask 2 (roles only).

other Message or Frame in the full task. To perform an evaluation, Messages from system and gold are assigned via linear sum assignment of the Message span's token overlap using form and STWR as tie-breakers. Each Message can only be matched to at most one other Message. The tie-breakers are needed to correctly assign Messages in rare cases as they can have the same offsets, yet use a different form or STWR. If a system predicts a Message that has no matching Message in the gold annotations, this increases the false positives for Message and each role system predicted as belonging to the unmatched Message. Vice versa, if a Message from the gold annotation has no match in the system prediction, the false negatives are increased. A correctly matched Message yields true positives for all correct roles according to the fraction of overlap and false negatives resp. false positives for tokens that were not identified resp. wrongly predicted by the system.

5.5 Baseline results

5.5.1 Task 1 – Parliamentary debates

Table 8 shows results for the baseline system (Task 1). The simple string match for the prediction of cues has a recall over 80% but precision is rather low with 57%. The heuristics-based role prediction thus suffers from error propagation (precision: 67%) and even more from the low coverage of our heuristic rules (recall: 32%). When applying the role prediction baseline to gold cues, we can see a substantial improvement for precision (89%) but not for recall.

A qualitative error analysis showed that, in addition to the low recall, many errors are due to incorrect syntactic parses. The dependency parser struggles with the long sentences and many parenthetical remarks included in the debates and, in addition, often fails to return the correct analysis for copula constructions.

5.5.2 Task 2 – News articles

Table 9 resp. Table 10 shows the results for the baseline system (Task 2) on the development resp. test set. The rule-based system is not tuned on the development set (and in fact not even trained on the training set). Consequently, there is almost no difference between the scores on the test and development set.

The results show that the system achieves decent precision while clearly suffering from low recall. The low recall mainly results from two causes. First, the system is not capable of predicting certain types of speech (Reported and Free Indirect) or roles (Addressee) that are present in the dataset. Second, the system was designed to prefer quality over quantity when automatically extracting quotations from large amounts of raw text. As such, the system has a preference for precision over recall even for types of speech that it can predict.

When comparing the results of the full task with the simplified task, it can be seen that the system has worse Message precision but slightly better Message recall. This phenomenon can be attributed to the fact that the system produces the same output for both subtasks – it does not differentiate between the tasks. Since it predicts some cases of nested Messages (e.g. Direct speech within Indirect speech) the Message precision on the simplified task (that does not include nesting) is lower. As a side effect recall is slightly increased

	Prec	Rec	F1
Subtask 1 (full task)			
Message	75.12	36.13	48.79
Roles	55.03	25.53	34.87
Joint	60.65	28.65	38.91
Form	57.78	29.56	39.11
STWR	56.59	28.94	38.30
Subtask 2 (simplified task)			
Message	71.29	36.46	48.25
Source	57.76	24.93	34.83
Joint	64.65	30.90	41.82

Table 9: Task 2 baseline results on the development set

	Prec	Rec	F1
Subtask 1 (full task)			
Message	70.75	36.22	47.91
Roles	55.60	26.05	35.48
Joint	59.86	28.99	39.06
Form	63.48	33.59	43.93
STWR	52.46	27.76	36.31
Subtask 2 (simplified task)			
Message	68.74	37.01	48.12
Source	53.98	22.47	31.73
Joint	61.56	30.02	40.36

Table 10: Task 2 baseline results on the test set

because in the reference data some instances of unsupported types of speech are excluded due to nesting. According to the joint score, the system performs better on the simplified task than the full task – while performing worse on MESSAGES. The reason for this is the averaging over all correct resp. predicted spans: In the simplified task, there is only a single role (SOURCE) and thus fewer role spans than in the full task. As the system is significantly better at predicting the MESSAGES than the roles, the joint performance increases on the simplified task.

5.6 Results of the SpkAtt2023 shared task

5.6.1 Task 1

The shared task had three participating teams that submitted their system results. Only two of the participating teams submitted a system description. Below we summarize the main features of each system. For details, see the system descriptions (Ehrmanntraut et al., 2023; Bornheim et al., 2023).

Speaker attribution with BERT The winning system is based on a large BERT model (deepset/gbert-large, Chan et al. (2020)) and divides the task into three subtasks. In the first step, the system tries to identify the cue words. Next, individual cue words are grouped into cue spans (i.e., multi-word cues) that trigger the same speech event. In the last step, given a group of cue words, the system predicts the associated roles for this cue as a multi-label classification task on the token level. To increase efficiency, the system does not fine-tune the full model parameters but inserts Low Rank Adapters (LoRA) (Hu et al., 2021) into the model that are then fine-tuned on the data, either in a token classification setup (cue word detection; role detection) or in a sequence classification task (detection of multi-word cues).

The participants also experimented with domain adaptation via continual pre-training on in-domain data but could not further improve their results.

Speaker attribution with Llama 2 The second-ranked system decided on a very different design for the speaker attribution task, using a prompt-based approach. The system is based on two fine-tuned Llama 2 models (Llama 2 70B) (Touvron et al., 2023), one for identifying the cues and one for role prediction. To reduce memory usage and make the system more efficient, QLoRA (Quantized Low-Rank Adaptation) (Dettmers et al., 2023) has been applied to quantize the model weights to four bits. Additionally, LoRA adapters are added to all linear transformer blocks of the model.

The prediction of cues and roles is done separately by means of two prompting mechanisms and postprocessing, in order to convert the system output into structured predictions for evaluation. More details on the implementation can be found in the system description (Bornheim et al., 2023).

Results A summary of the results can be seen in Table 8. All three systems beat the joint baseline for both, subtask 1 and 2. While the two best-ranked systems yield very similar results for cue prediction, the BERT-based system clearly outperforms the QLoRA-adapted Llama 2 model for role prediction with regard to recall (82% vs. 87%).

Interestingly, for role prediction on *automatically predicted* cues the QLoRA-adapted LLM seems to outperform the BERT-based system.[11]

[11] But note that the results are based on one run only, with no information on the standard deviation of the systems. Thus,

When predicting roles on gold cues, however, this advantage disappears and the BERT-based system beats the other systems in both, precision and recall.

5.6.2 Task 2

Since no team submitted an official run for Task 2, the only results on this task are the baseline results presented in Section 5.5.2. Thus, we are looking forward to task and dataset being used in future experiments and evaluations.

6 Conclusions

We presented an overview of the GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates. The shared task provided two new datasets, one including parliamentary debates from the German Bundestag (Task 1) and one from the news domain (Task 2). Each task consisted of two subtasks. All data is made available, either via a GitHub repository (train and dev sets) or in codalab (test sets for evaluation).

The outcome of the shared task showed results close to 90% F1 for the detection of cue words and well above 80% F1 for role prediction on automatically predicted cues (Task 1). When also providing the gold cues, we see a further increase in results for role prediction up to 90% F1. The high accuracy of the results should enable new applications in the computational social sciences and the release of the new datasets will provide the basis for further improvements for speaker attribution in German text.

Acknowledgements

We would like to thank the German Society for Computational Linguistics and Speech Technology (GSCL) for their generous financial support for the creation of the Task 1 dataset. We also thank our annotators for their dedicated work. Part of this project has been supported by a research grant from the Ministry of Science, Research and the Arts (MWK) Baden-Württemberg.

References

Mariana S. C. Almeida, Miguel B. Almeida, and André F. T. Martins. 2014. A joint model for quotation attribution and coreference resolution. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 39–48.

Thomas Bögel and Michael Gertz. 2015. Did I really say that? - Combining machine learning and dependency relations to extract statements from German news articles. In *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology, GSCL 2015, University of Duisburg-Essen, Germany, 30th September - 2nd October 2015*, pages 13–21. GSCL e.V.

Tobias Bornheim, Niklas Grieger, Patrick Gustav Blaneck, and Stephan Bialonski. 2023. Speaker Attribution in German Parliamentary Debates with QLoRA-adapted Large Language Models. In *The GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates, co-located with KONVENS 2023*, Ingolstadt, Germany.

Annelen Brunner. 2015. Automatic recognition of speech, thought, and writing representation in german narrative texts. *Literary and Linguistic Computing*, 28(4):563 – 575.

Annelen Brunner, Stefan Engelberg, Fotis Jannidis, Ngoc Duyen Tanja Tu, and Lukas Weimer. 2020. Corpus redewiedergabe. In *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC), May 11-16, 2020, Palais du Pharo, Marseille, France*, pages 803 – 812, Paris. European Language Resources Association.

Annelen Brunner, Ngoc Duyen Tanja Tu, Lukas Weimer, and Fotis Jannidis. 2019. Deep learning for free indirect representation. In *Preliminary proceedings of the 15th Conference on Natural Language Processing (KONVENS 2019), October 9 – 11, 2019 at Friedrich-Alexander-Universität Erlangen-Nürnberg*, pages 241 – 245, München [u.a.]. German Society for Computational Linguistics & Language Technology und Friedrich-Alexander-Universität Erlangen-Nürnberg.

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005. Identifying sources of opinions with conditional random fields and extraction patterns. In *Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, HLT/EMNLP 2005, pages 355–362.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs.

more evaluation is required before we can draw any final conclusions.

Tillmann Dönicke, Hanna Varachkina, Anna Mareike Weimer, Luisa Gödeke, Florian Barth, Benjamin Gittel, Anke Holler, and Caroline Sporleder. 2022. Modelling Speaker Attribution in Narrative Texts With Biased and Bias-Adjustable Neural Networks. *Frontiers in Artificial Intelligence*, 4:725321.

Anton Ehrmanntraut, Leonard Konle, and Fotis Jannidis. 2023. Politics, BERTed: Automatic Attribution of Speech Events in German Parliamentary Debates. In *The GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates, colocated with KONVENS 2023*, Ingolstadt, Germany.

David K. Elson and Kathleen R. McKeown. 2010. Automatic attribution of quoted speech in literary narrative. In *The Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI 2010.

Susanne Fertmann. 2016. Using speaker identification to improve coreference resolution in literary narratives. Master's thesis, Computational Linguistics.

Yulia Grishina and Manfred Stede. 2017. Multi-source projection of coreference chains: assessing strategies and testing opportunities. In *The 2nd Coreference Resolution Beyond OntoNotes Workshop*, CORBON-2017.

Hua He, Denilson Barbosa, and Grzegorz Kondrak. 2013. Identification of speakers in novels. In *The 51st Annual Meeting of the Association for Computational Linguistics*, ACL 2013, pages 1312–1320.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, Adriane Boyd, et al. 2020. spacy: Industrial-strength natural language processing in python.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *CoRR*, abs/2106.09685.

Richard Johansson and Alessandro Moschitti. 2013. Relational features in fine-grained opinion analysis. *Computational Linguistics*, 39(3):473–509.

Evgeny Kim and Roman Klinger. 2018. Who feels what and why? annotation of a literature corpus with semantic roles of emotions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1345–1359, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The INCEpTION platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9, Santa Fe, New Mexico. Association for Computational Linguistics.

Ralf Krestel, Sabine Bergler, and René Witte. 2008. Minding the source: Automatic tagging of reported speech in newspaper articles. In *The International Conference on Language Resources and Evaluation*, LREC 2008.

Markus Krug, Frank Puppe, Isabella Reger, Lukas Weimer, Luisa Macharowsky, and Stephan Feldhaus. 2018. *Description of a Corpus of Character References in German Novels – DROC [Deutsches ROman Corpus]. DARIAH-DE Working Papers. Göttingen: DARIAH-DE.*

Ana Marasovic and Anette Frank. 2018. SRL4ORL: improving opinion role labeling using multi-task learning with semantic role labeling. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 583–594. Association for Computational Linguistics.

Grace Muzny, Angel X. Chang, Michael Fang, and Dan Jurafsky. 2017. A two-stage sieve approach for quote attribution. In *The 15th Conference of the European Chapter of the Association for Computational Linguistics*, EACL 2017, pages 460–470.

Sean Papay and Sebastian Padó. 2020. RiQuA: A corpus of rich quotation annotation for English literary text. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 835–841, Marseille, France. European Language Resources Association.

Silvia Pareti. 2015. *Attribution: a computational approach*. Ph.D. thesis, University of Edinburgh, UK.

Silvia Pareti, Timothy O'Keefe, Ioannis Konstas, James R. Curran, and Irena Koprinska. 2013. Automatically detecting and attributing indirect quotations. In *The 2013 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2013, pages 989–999.

Bruno Pouliquen, Ralf Steinberger, and Clive Best. 2007. Automatic detection of quotations in multilingual news. In *The International Conference on Recent Advances in Natural Language Processing*, RANLP 2007, pages 487–492.

Josef Ruppenhofer, Caroline Sporleder, and Fabian Shirokov. 2010. Speaker attribution in cabinet protocols. In *The Seventh conference on International Language Resources and Evaluation*, LREC 2010.

Christian Scheible, Roman Klinger, and Sebastian Padó. 2016. Model architectures for quotation detection. In *The 54th Annual Meeting of the Association for Computational Linguistics*, ACL 2016.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Michael Wiegand and Dietrich Klakow. 2012. Generalization methods for in-domain and cross-domain opinion holder extraction. In *The 13th Conference of the European Chapter of the Association for Computational Linguistics*, EACL 2012, pages 325–335.

Appendix

A1	ADDRESSEE	EVIDENCE	MEDIUM	MESSAGE	PARTICLE	SOURCE	TOPIC	NONE
ADDRESSEE	679	0	0	26	7	7	14	279
EVIDENCE	0	90	11	0	0	0	0	23
MEDIUM	0	64	109	17	0	5	18	245
MESSAGE	42	25	27	11,734	7	52	662	3,570
PARTICLE	0	0	0	8	101	0	1	46
SOURCE	22	15	8	106	1	2,244	22	623
TOPIC	4	3	0	214	0	0	574	194
NONE	310	116	48	3,530	91	407	335	0

Table 11: Confusion matrix (token level) for role annotations for the last two annotation samples (Task 1).

Speaker Attribution in German Parliamentary Debates with QLoRA-adapted Large Language Models

Tobias Bornheim[1], Niklas Grieger[1,2,3], Patrick Gustav Blaneck[1] and Stephan Bialonski[1,3,*]

[1] *Department of Medical Engineering and Technomathematics*
FH Aachen University of Applied Sciences, Jülich, Germany

[2] *Department of Information and Computing Sciences*
Utrecht University, Utrecht, The Netherlands

[3] *Institute for Data-Driven Technologies*
FH Aachen University of Applied Sciences, Jülich, Germany
[*] *bialonski@fh-aachen.de*

Abstract

The growing body of political texts opens up new opportunities for rich insights into political dynamics and ideologies but also increases the workload for manual analysis. Automated speaker attribution, which detects who said what to whom in a speech event and is closely related to semantic role labeling, is an important processing step for computational text analysis. We study the potential of the large language model family Llama 2 to automate speaker attribution in German parliamentary debates from 2017–2021. We fine-tune Llama 2 with QLoRA, an efficient training strategy, and observe our approach to achieve competitive performance in the GermEval 2023 Shared Task On Speaker Attribution in German News Articles and Parliamentary Debates. Our results shed light on the capabilities of large language models in automating speaker attribution, revealing a promising avenue for computational analysis of political discourse and the development of semantic role labeling systems.

1 Introduction

Language is central to the study of politics, as it forms the basis for political speech and debates (Grimmer and Stewart, 2013). These textual sources offer rich insights into political dynamics and ideologies, yet the analysis of even moderately sized collections has been impeded by prohibitive costs. Recent innovations from natural language processing (NLP) have the potential to significantly reduce the financial burden of scrutinizing extensive text corpora (Glavaš et al., 2019; Abercrombie and Batista-Navarro, 2020). This development coincides with the availability of a growing body of political texts, including German Parliamentary data (Barbaresi, 2018; Blätte and Blessing, 2018; Walter et al., 2021; Rauh and Schwalbach, 2020; Abrami et al., 2022; Rehbein et al., 2023), thus opening new avenues for political research.

Political texts are usually unstructured, presenting challenges for automated analyses. An approach towards this challenge is automated speaker attribution (Rehbein et al., 2023), which detects who said what to whom in a speech event. This process involves detecting cue words that initiate a speech event and discerning the different roles (e.g., source, message, and addressee) associated with each event. This task is closely related to semantic role labeling (SRL) that delineates the specific semantic relationships among a predicate and its corresponding arguments, such as “who” did “what” to “whom,” “where,” “when,” and “why” (Gildea and Jurafsky, 2002; Màrquez et al., 2008). Semantic role labeling is considered a key component for natural language understanding and has been demonstrated to enhance systems for various applications including question answering, machine translation, and video understanding (Navigli et al., 2022).

Early approaches to SRL relied on syntactic features (Navigli et al., 2022; Larionov et al., 2019). More recently, the field has seen a significant transition from such engineered features to features learned in an end-to-end fashion by models that operate on raw-level input or tokens (Collobert et al., 2011). However, such end-to-end models necessitate large annotated training sets, available for English but scarce for low-resource languages. This problem can be mitigated by pretraining on unannotated data. Indeed, the emergence of pretrained large language models (LLMs) inspired by the transformer architecture (Vaswani et al., 2017) led to new state-of-the-art results across various NLP tasks. Among these, encoder-only models like BERT were demonstrated to improve existing SRL benchmarks (Shi and Lin, 2019). More recently, the advent of decoder-only models, such as GPT (Radford and Narasimhan, 2018) and larger models like GPT-4 (OpenAI, 2023), Claude 2 (Bai et al., 2022), and Llama 2 (Touvron et al., 2023b), has further propelled the field. These models, with their ability to comprehend and execute instructions in natural language for a wide array of tasks, hold potential for SRL and automated speaker attribution that is, to the best of our knowledge, largely unexplored.

In this contribution, we study the potential of Llama 2 70B, a model from a recently introduced family of large language models, to automatically detect speech events and attribute speakers in German parliamentary debates. We instruct and fine-tune Llama 2 to extract cues and roles using QLoRA (Dettmers et al., 2023), a parameter- and computationally efficient training strategy. Our approach achieves competitive performance (quantified by F1 scores for cues and roles) on the SpkAtt-2023 dataset of the *GermEval 2023 Shared Task on Speaker Attribution in German News Articles and Parliamentary Debates* (Rehbein et al., 2023). The implementation details of our experiments (Team “CPAa”) are available online[1].

2 Data and tasks

The dataset of the *GermEval 2023 Shared Task on Speaker Attribution in German News Articles and Parliamentary Debates* consisted of 267 speeches from the German Bundestag (Rehbein et al., 2023). This dataset included speeches from all seven parliamentary groups (including independent members of parliament as a separate group) of the 19th legislative period of the German Bundestag (see Table 1 for details). To facilitate analysis, each speech was automatically separated into sentence-like structures using spaCy, hereafter referred to as *samples* (units of analysis). Each sample was then further split into *elements*, i.e., words and punctuation marks.

Human annotators followed annotation guidelines (Rehbein et al., 2023) to assign none, one, or multiple annotations to each sample. These annotations consisted of *cue words* that invoke speech events and roles (*Addr*, *Evidence*, *Medium*, *Message*, *Source*, *Topic*, *PTC*) associated with that event. While the cue is mandatory for each annotation, roles are context-dependent and may be absent. Figure 1 shows example annotations.

Parliamentary group	*Speeches*	*Samples*
CDU/CSU	77	4305
SPD	57	2887
AfD	39	1827
FDP	34	1435
DIE LINKE	29	1356
B’90 / DIE GRÜNEN	27	1152
independent	4	125
Total	267	13087

Table 1: Number of speeches and samples per parliamentary group in the combined *Train*, *Dev*, and *Eval* datasets.

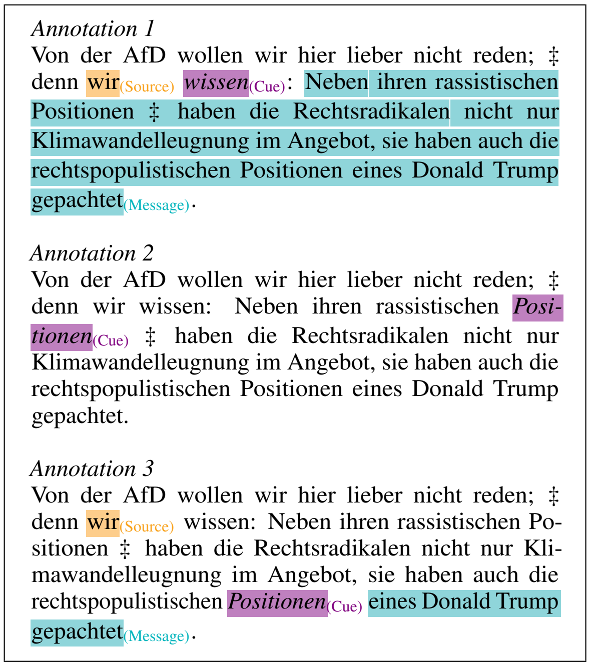
Split	*Speeches*	*Samples*	*Annotations*
Dev	18	927	515
Train	177	9093	5399
Eval	72	3067	1792
Total	267	13087	7706

Table 2: Number of speeches, samples (units of analysis), and annotations for each dataset. The *Trial* dataset is completely contained within the *Train* dataset and is therefore not shown. The *Eval* dataset here refers to the test sets of both *Subtask 1* and *Subtask 2*, since they only differ in the provided annotations.

The Shared Task consisted of two subtasks: *Full Annotation* (*Subtask 1*) and *Role Detection* (*Subtask 2*) (Rehbein et al., 2023). In the *Full Annotation* subtask, the goal was to predict all cues and roles for each sample. In the *Role Detection* subtask, the gold cues were given, and the goal was to predict only the roles for each sample.

The dataset was provided as five sets, namely *Trial*, *Train*, *Dev*, and two *Eval* sets, one for each subtask, see Table 2. We omitted the *Trial* dataset in our experiments, since it was included in the *Train* dataset. For training and tuning the final models, we used the *Train* and *Dev* datasets. The two *Eval* datasets were only used to compute the final scores of the shared task.

[1] https://github.com/dslaborg/germeval2023



Annotation 1
Von der AfD wollen wir hier lieber nicht reden; ‡ denn wir(Source) *wissen*(Cue): Neben ihren rassistischen Positionen ‡ haben die Rechtsradikalen nicht nur Klimawandelleugnung im Angebot, sie haben auch die rechtspopulistischen Positionen eines Donald Trump gepachtet(Message).

Annotation 2
Von der AfD wollen wir hier lieber nicht reden; ‡ denn wir wissen: Neben ihren rassistischen *Positionen*(Cue) ‡ haben die Rechtsradikalen nicht nur Klimawandelleugnung im Angebot, sie haben auch die rechtspopulistischen Positionen eines Donald Trump gepachtet.

Annotation 3
Von der AfD wollen wir hier lieber nicht reden; ‡ denn wir(Source) wissen: Neben ihren rassistischen Positionen ‡ haben die Rechtsradikalen nicht nur Klimawandelleugnung im Angebot, sie haben auch die rechtspopulistischen *Positionen*(Cue) eines Donald Trump gepachtet(Message).

Figure 1: Sentence from the *Train* dataset with three annotations. The sentence was split into three samples by spaCy (splitting points are indicated by ‡). As seen in *Annotation 2*, there can be annotations consisting of only cue word(s). *Annotation 1* and *Annotation 3* show that annotated roles can span multiple samples.

3 Methods

3.1 Models

We used the Llama 2 model family (Touvron et al., 2023b), a set of large language models pretrained on a corpus of two trillion tokens with a context length of 4096 tokens. The Llama 2 model family includes both pretrained models and fine-tuned versions optimized for conversational tasks. Since our approach did not require the conversational capabilities of the fine-tuned models, we chose to use the base pretrained versions of Llama 2 in our experiments. These base models were trained without a specific prompt format and are therefore not biased toward any particular prompt strategy, allowing us to freely choose our own prompt format.

While the Llama 2 model family contains models of various sizes, we chose to fine-tune the largest available model with 70 billion parameters (Llama 2 70B). The weights of this model can be obtained upon request using the official GitHub repository[2]. Once downloaded, we followed the provided instructions[3] to convert the model to the HuggingFace Transformers format (Wolf et al., 2020). This conversion allowed us to load the model using the HuggingFace Transformers library, which facilitated the fine-tuning and inference steps.

3.2 Preprocessing

For effective training (see section 3.3) and inference (see section 3.4) we preprocessed each sample. We parsed each annotation into its respective lists of elements. Next, we joined all elements of a sample with space characters in between to get each sample's *text*. Since roles can be contained in samples different from the one containing the cue, we concatenated the sample with the next two samples of the same speech, if possible.

During our experiments, we noticed that our models ignored their instructions and generated random text if the text of a given sample ended with a colon. To counteract this behavior, we replaced this trailing colon with a period.

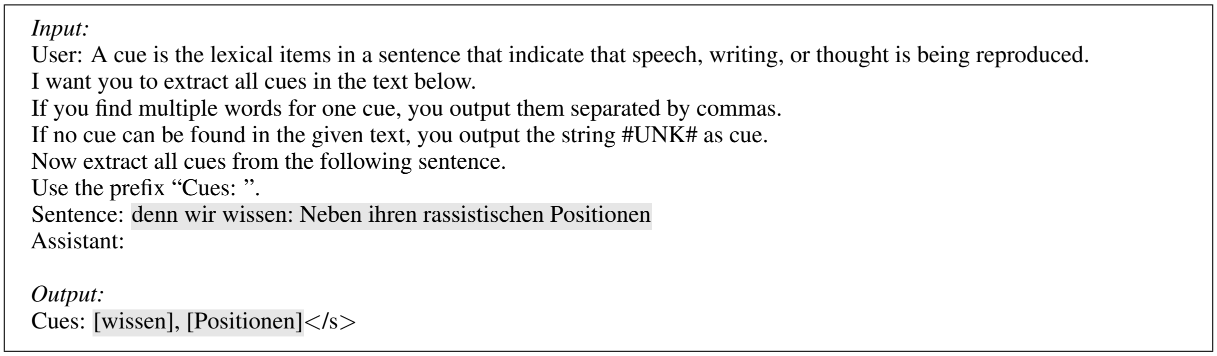
We designed prompts for cue prompting (see Figure 2) and role prompting (see Figure 3). We wrote the instructions in our prompt templates in English, because it was observed that the performance of multilingual models such as Llama 2 is improved when English prompts are used (Fu et al., 2022; Huang et al., 2023). Also, since a sample may not contain a cue, or a role may be missing, we used "#UNK#" to mark such cases.

3.3 Training

For our final submission, we fine-tuned two Llama 2 70B models to identify cues and roles, respectively, using QLoRA (Quantized Low-Rank Adaptation) (Dettmers et al., 2023). QLoRA is a highly efficient fine-tuning technique for large language models that achieves similar performance to full fine-tuning while using only a fraction of the memory. This memory reduction is achieved by quantizing the model weights of an LLM to four bits and adding Low Rank Adapters (LoRA layers) to all linear transformer blocks of the model. During fine-tuning, only these LoRA layers are trained and the rest of the pretrained model weights remain unaltered. By employing this strategy, QLoRA achieves a significant reduction in memory usage during fine-tuning, while still allowing the model

[2] https://github.com/facebookresearch/llama

[3] https://github.com/facebookresearch/llama-recipes



Input:
User: A cue is the lexical items in a sentence that indicate that speech, writing, or thought is being reproduced.
I want you to extract all cues in the text below.
If you find multiple words for one cue, you output them separated by commas.
If no cue can be found in the given text, you output the string #UNK# as cue.
Now extract all cues from the following sentence.
Use the prefix “Cues: ”.
Sentence: denn wir wissen: Neben ihren rassistischen Positionen
Assistant:

Output:
Cues: [wissen], [Positionen]</s>

Figure 2: Example cue prompt and desired model response for the sample “denn wir wissen: Neben ihren rassistischen Positionen” with the cues “wissen” and “Positionen”. Shaded in gray are the parts of the prompt and response that are sample dependent. The prompt is used as the *Input* sequence for training and inference, while the *Output* sequence contains the desired response with the cues. The end-of-sentence token “</s>” is used to indicate the end of the *Output* sequence.

to adapt to downstream tasks through the trainable LoRA layers.

As described in Section 3.2, we parsed the training samples into cue prompts (see Figure 2) that served as input to the cue model and role prompts (see Figure 3) that served as input to the role model. The models were not fine-tuned on these input sequences, but rather on the desired assistant responses (defined as *Output* in Figures 2 and 3). This approach is consistent with previous research that has shown improved performance when fine-tuning only on the target response of an instruction set, rather than both the instructions and the desired response (Dettmers et al., 2023). By treating the input and output separately, we can process the two sequences with different maximum sequence lengths. Specifically, for the model used to identify cues, we set the maximum length of the input to 256 tokens (with seven samples of the training data truncated) and the maximum length of the output to 64 tokens (no samples truncated). For the model used to identify roles, we truncated the input to 640 tokens (with six samples of the training data truncated) and the output to 256 tokens (with one sample truncated).

Except for the maximum number of tokens in the input and output sequences, we largely followed the training strategy proposed by Dettmers et al. (Dettmers et al., 2023). Although their specific experiments did not involve a Llama 2 70B model, they successfully fine-tuned a similarly sized LLaMA model (predecessor to Llama 2) with 65 billion parameters (Touvron et al., 2023a). We adopted most parameters from this 65B model fine-tuning, such as a constant learning rate of $\eta = 0.0001$ with linear warmup over the first 3% of training steps and a dropout of 0.05 for the LoRA layers. The main hyperparameter we adjusted was the number of training steps to prevent overfitting. For the cues model, we trained for 2000 steps with a batch size of 16 and no gradient accumulation. For the roles model, we used 2500 steps with a batch size of eight and gradient accumulation over two steps, i.e., an effective batch size of 16.

Fine-tuning was carried out on a DGX A100 server, with a total training time of about seven hours for the cues model and 17 hours for the roles model. To optimize memory usage, we experimented with reducing the batch size to one while increasing the gradient accumulation steps to 16 (i.e., maintaining the same effective batch size). With these parameters, both models were able to operate within a GPU memory limit of less than 60 GB.

3.4 Inference

Prompting our fine-tuned models was a two-step process. In the first step, we prompted our cue model for all cues in a sample using our prompt template for cues (see Figure 2). We postprocessed the output of the model (see section 3.5) into a list of cues. In the second step, for each cue, we prompted for the roles with our role model. To do this, we prepended the complete cue prompt and its output to the role prompt template before querying the model (see Figure 3).

To ensure reproducibility of results, we configured our models to generate output deterministically. For a given input sequence, large language models obtain a probability distribution over all

Input:
User: Now I give you again the sentence only in addition with the two following sentences, because the roles can be partially contained in the following sentences.
Text: denn wir wissen : Neben ihren rassistischen Positionen ‡ haben die Rechtsradikalen nicht nur Klimawandelleugnung im Angebot , sie haben auch die rechtspopulistischen Positionen eines Donald Trump gepachtet . ‡ Als Linke übernehmen wir Verantwortung .
Now find all roles in the sentence associated with the cue ‘wissen’ you found in the beginning sentence.
Assistant:

Output:
cue: wissen
ptc: #UNK#
evidence: #UNK#
medium: #UNK#
topic: #UNK#
addr: #UNK#
message: Neben, ihren, rassistischen, Positionen, haben, die, Rechtsradikalen, nicht, nur, Klimawandelleugnung, im, Angebot, ,, sie, haben, auch, die, rechtspopulistischen, Positionen, eines, Donald, Trump, gepachtet
source: wir</s>

Figure 3: Example role prompt and desired model response for the sample “denn wir wissen: Neben ihren rassistischen Positionen” with the cue “wissen”. Since roles can be contained in samples different from the one containing the cue, we concatenated the sample with the next two samples of the same speech (transitions between samples are indicated by ‡). Shaded in gray are the parts of the prompt and response that are sample dependent. Similar to the cue prompt, the role prompt is used as the *Input* sequence for training and inference, while the *Output* sequence contains the desired response. We append the end-of-sentence token “</s>” to the *Output*.

possible tokens. We chose to always select the token with the highest assigned probability as the next output token, thereby fixing the output for a given input sequence.

3.5 Postprocessing and evaluation metrics

Several postprocessing steps were necessary to evaluate the models’ output in a structured way.

Enforcing the output format. If the models’ output did not follow our strict output format (see Figures 2 and 3), we mapped the output to the marker #UNK# (unknown).

Preventing overlapping cues. If our cue model detected multiple but overlapping cues, we combined them into a single cue.

Ignoring made-up words. If the output of the model contained words for cues or roles that were not in the given sample, and no other word with a Levenshtein distance of 1 was found in the sample, we ignored those words. Then, if the output was empty, we mapped the output to the marker #UNK# (unknown).

Resolving ambiguities. A word may occur more than once in a sample. When a model outputs such a word as a cue or a role, it is unclear to which occurrence of the word in the sample it should be attributed. To resolve this ambiguity, for each occurrence of the word, we counted how many elements around that word (in the range of two elements to the left and right) were part of the cue or role, and chose the occurrence with the highest count.

Including surrounded punctuation. Roles often contained punctuation marks such as colons or commas. We observed that our models ignored these punctuation marks most of the time. If a punctuation mark was surrounded by words that were selected for this role, we added that punctuation mark to the role as well.

Evaluating metrics. To evaluate the performance of our models, we used the proportional F1 score as proposed for opinion role labeling (Johansson and Moschitti, 2010). This score is defined as the harmonic mean of the proportional precision and recall. Proportional precision quantifies the proportion of overlap between a predicted cue (role) and an overlapping true cue (role). Proportional recall quantifies the proportion of overlap between a true cue (role) and an overlapping predicted cue (role; see (Rehbein et al., 2023) for further details on how the proportional F1 score is calculated).

	Precision	*Recall*	*F1*
Subtask 1			
Cues	0.889	0.889	0.889
Roles	0.787	0.822	0.804
Cues & Roles	0.798	0.829	0.813
Subtask 2			
Roles	0.910	0.873	0.891

Table 3: Proportional precision, recall, and F1 scores obtained for predicting cues and roles on the *Eval* dataset. The joint scores for predicting both cues and roles (Subtask 1 of GermEval 2023 Shared Task 1) are shown in the third row. The last row shows the results obtained for predicting roles on the *Eval* dataset when the true cues were given (Subtask 2).

4 Results

We used the same fine-tuned Llama 2 70B models for both Subtask 1 and Subtask 2 of GermEval 2023 Shared Task 1 – a cues model to identify cues in a given sentence and a roles model to predict the roles associated with the identified cues. While the cues model was used exclusively in Subtask 1, as the cues were provided in Subtask 2, the roles model was used in both subtasks. It leveraged either the predicted cues from Subtask 1 or the gold cues from Subtask 2 to predict the roles associated with each cue, as described in section 3.4. By using the same fine-tuned roles model for both subtasks, we were able to analyze the impact of using gold cues versus predicted cues on role identification performance.

Table 3 shows the final results of our submissions on the *Eval* dataset, as reported by the organizers of the GermEval 2023 Shared Task. For Subtask 1, the fine-tuned cues model achieved an F1 score of 0.889 for predicting cues. Using the predicted cues from this model, the fine-tuned roles model achieved an F1 score of 0.804 for predicting roles. Combining both predictions, our models achieved an overall F1 score of 0.813 for predicting cues and roles in Subtask 1. In Subtask 2, where gold cues were provided, the same roles model used in Subtask 1 achieved a higher F1 score of 0.891 for predicting roles. Interestingly, the improvement of the roles model using gold cues was greater in precision, which increased from 0.787 to 0.910, than in recall, which increased from 0.822 to 0.873. This increase in precision suggests that the cues model in Subtask 1 overpredicted sentences as containing cues when they actually had no cues, resulting in too many false positive role predictions.

In summary, our results demonstrate that our fine-tuned models are effective at reliably predicting cues and roles. Additionally, the results highlight the importance of accurate cue prediction, as errors of the cues model propagate to the roles model, reducing its performance.

5 Conclusion

We demonstrated that fine-tuned Llama 2 language models can successfully predict cues and roles in German parliamentary debates, achieving competitive performance on the GermEval2023 Shared Task without relying on traditional linguistic features. These results highlight the feasibility of automated speaker attribution by fine-tuning models on prompt templates that task them with identifying cues and roles. The similarity between automated speaker attribution and semantic role labeling suggests that this strategy may pave the way for new state-of-the-art results in various semantic role labeling tasks.

Limitations

We did not study risks that may or may not arise when our fine-tuned large language models are used for other application scenarios than ours. In our approach, users can neither manipulate the prompts nor read the generated texts produced by our models. Instead, the generated outputs are processed and mapped back to the words from the parliamentary speeches used as input. Therefore, we consider the risks associated with our approach to be limited. We recommend security testing if our trained models are to be used in other scenarios.

Acknowledgements

We are grateful to M. Reißel and V. Sander for providing us with computing resources.

References

Gavin Abercrombie and Riza Batista-Navarro. 2020. Sentiment and position-taking analysis of parliamentary debates: A systematic literature review. *Journal of Computational Social Science*, 3(1):245–270.

Giuseppe Abrami, Mevlüt Bagci, Leon Hammerla, and Alexander Mehler. 2022. German parliamentary corpus (GerParCor). In *Proceedings of the Thirteenth Language Resources and Evaluation Conference, LREC 2022, Marseille, France, 20-25 June*

2022, pages 1900–1906. European Language Resources Association.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosiute, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemí Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022. Constitutional AI: Harmlessness from AI feedback. *CoRR*, abs/2212.08073.

Adrien Barbaresi. 2018. A corpus of German political speeches from the 21st century. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Andreas Blätte and André Blessing. 2018. The GermaParl corpus of parliamentary protocols. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA).

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. 2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. *CoRR*, abs/2305.14314.

Jinlan Fu, See-Kiong Ng, and Pengfei Liu. 2022. Polyglot Prompt: Multilingual multitask prompt training. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9919–9935, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Comput. Linguistics*, 28(3):245–288.

Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. 2019. Computational analysis of political texts: Bridging research efforts across communities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*. Association for Computational Linguistics.

Justin Grimmer and Brandon M. Stewart. 2013. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, 21(3):267–297.

Haoyang Huang, Tianyi Tang, Dongdong Zhang, Wayne Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in LLMs: Improving multilingual capability by cross-lingual-thought prompting. *CoRR*, abs/2305.07004.

Richard Johansson and Alessandro Moschitti. 2010. Syntactic and semantic structure for opinion expression detection. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 67–76, Uppsala, Sweden. Association for Computational Linguistics.

Daniil Larionov, Artem Shelmanov, Elena Chistova, and Ivan V. Smirnov. 2019. Semantic role labeling with pretrained language models for known and unknown predicates. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing, RANLP 2019, Varna, Bulgaria, September 2-4, 2019*, pages 619–628. INCOMA Ltd.

Lluís Màrquez, Xavier Carreras, Kenneth C. Litkowski, and Suzanne Stevenson. 2008. Semantic role labeling: An introduction to the special issue. *Comput. Linguistics*, 34(2):145–159.

Roberto Navigli, Edoardo Barba, Simone Conia, and Rexhina Blloshmi. 2022. A tour of explicit multilingual semantics: Word sense disambiguation, semantic role labeling and semantic parsing. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing: Tutorial Abstracts*, pages 35–43, Taipei. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pretraining.

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.

Ines Rehbein, Fynn Petersen-Frey, Annelen Brunner, Josef Ruppenhofer, Chris Biemann, and Simone Paolo Ponzetto. 2023. Overview of the GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates. In *The GermEval 2023 Shared Task at KONVENS 2023*, Ingolstadt, Germany.

Peng Shi and Jimmy Lin. 2019. Simple BERT models for relation extraction and semantic role labeling. *CoRR*, abs/1904.05255.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. LLaMA: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Annual Conf. Neural Information Processing Systems 2017*, pages 5998–6008, Long Beach, CA, USA.

Tobias Walter, Celina Kirschner, Steffen Eger, Goran Glavas, Anne Lauscher, and Simone Paolo Ponzetto. 2021. Diachronic analysis of German parliamentary proceedings: Ideological shifts through the lens of political biases. In *ACM/IEEE Joint Conference on Digital Libraries, JCDL 2021, Champaign, IL, USA, September 27-30, 2021*, pages 51–60. IEEE.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proc. 2020 Conf. on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Politics, BERTed: Automatic Attribution of Speech Events in German Parliamentary Debates

Anton Ehrmanntraut
Julius-Maximilans-Universität Würzburg
anton.ehrmanntraut@uni-wuerzburg.de

Abstract

This paper documents and analyzes a submission to the Shared Task on Speaker Attribution hosted at KONVENS 2023 (Rehbein et al., 2023). One task was the automatic identification of speech events in German parliamentary debates, i.e., where speech, thought or writing is referenced by speakers of parliament. The system approaches this with a token and sequence classification setup and offers a BERT-based solution to this task. According to the results, the proposed system performs surprisingly well despite its simple architecture. Further experiments indicate that even with a smaller variant of BERT, the system performs nearly equally well, whereas a domain adaptation of BERT on parliamentary speeches offered close to zero improvement.

1 Introduction

This paper presents a participating system at the *KONVENS 2023 Shared Task on Speaker Attribution (SpkAtt-2023)*, particularly participating in the task 1 on German parliamentary debates. The goal of the shared task is the automatic identification of speech events in political debates (whereas, for task 2, in news articles) and attributing them to their respective speakers, essentially identifying who says what to whom in the parliamentary debates (Rehbein et al., 2023). This is motivated by the fact that the automatic identification of such information is a prerequisite for an extensive semantic analysis of unstructured texts. For instance, the information automatically inferred from parliamentary speeches could be used for political discourse studies of parliamentary debates, or political communication.[1]

A *speech event* refers to a reference to speech, writing or thought by a member of parliament during one of their plenary speeches. Each such speech event consists of several word spans: first, a nonempty span of *cue words* that trigger this speech event (usually a verb), and second, several *role spans* associated with this speech event, i.e., *Source, Addressee, Message, Topic, Medium, Evidence*, or *Particles*, any of which can be empty, and all may pairwise overlap.[2] See Figure 1 for some examples. In this sense, a speech event does not have to be attributed to the actual person delivering the speech in parliament: the person may, for example, also state the thoughts of another entity, such as depicted in Figure 1(b).

The system presented in this paper approaches automatic speaker attribution through multiple fine-tuned BERT Transformer models (Devlin et al., 2019), designed to handle cue detection, cue linkage, and role detections. The system is specifically designed to be a minimal BERT-based baseline; all involved NLP tasks are essentially simple token classifications resp. sequence classifications. The model is similar to a semantic role labeling model by Shi and Lin (2019); in both models, entire sentences were encoded to leverage the contextual information from all tokens in the sequence at the same time.

The system was trained on the GePaDe dataset[3] for speaker attribution in German parliamentary debates, which has been specifically created for the SpkAtt-2023 task. It consists of 265 speeches, mostly from the 19th legislative term of the German Bundestag. For the shared task evaluations, the task organizers tested the submitted systems on (blind) test data. According to the official scorer, the presented system achieved a SpkAtt-F1 score of 0.83 on full inference (subtask 1a), and a SpkAtt-F1 score of 0.92 on a simplified task where gold cue

[1] See also the GePaDe datasheet: https://github.com/umanlp/SpkAtt-2023/blob/master/doc/SpkAtt-Debates-Datasheet.pdf.

[2] See also the precise annotation guidelines: https://github.com/umanlp/SpkAtt-2023/blob/master/doc/Guidelines_SpeakerAttribution_in_Parliamentary_Debates-SpkAtt-2023_Task1.pdf.

[3] https://github.com/umanlp/SpkAtt-2023

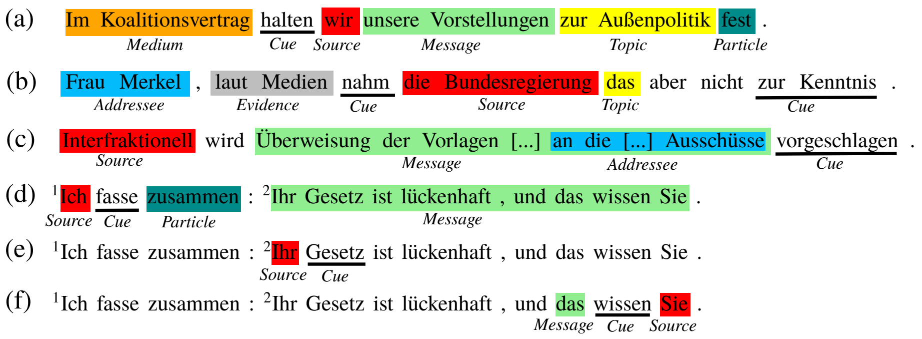

(a) Im Koalitionsvertrag halten wir unsere Vorstellungen zur Außenpolitik fest .
Medium Cue Source Message Topic Particle
(b) Frau Merkel , laut Medien nahm die Bundesregierung das aber nicht zur Kenntnis .
Addressee Evidence Cue Source Topic Cue
(c) Interfraktionell wird Überweisung der Vorlagen [...] an die [...] Ausschüsse vorgeschlagen .
Source Message Addressee Cue
(d) 1Ich fasse zusammen : 2Ihr Gesetz ist lückenhaft , und das wissen Sie .
Source Cue Particle Message
(e) 1Ich fasse zusammen : 2Ihr Gesetz ist lückenhaft , und das wissen Sie .
Source Cue
(f) 1Ich fasse zusammen : 2Ihr Gesetz ist lückenhaft , und das wissen Sie .
Message Cue Source

Figure 1: Example instances for the speaker attribution task. Note how the cue span can cover multiple tokens in a non-contiguous way (b). Note how, in the same speech event, words can be assigned to multiple role spans (c; from the GePaDe training set ID197411900). Also note how two annotations may overlap, how annotations may span multiple sentences (d and e), and how multiple annotations can be present even in the same sentence (e and f).

words are already given (subtask 1b). The entire system is made available.[4]

2 Related Work

Speaker attribution has many parallels to semantic role labeling. Similar to speaker attribution, semantic role labeling refers to the task of identifying the predicate of a clause, establishing "what" took place (typically a verb) and the associated arguments that specify the "who," the "what," "where," etc. Like the speaker attribution system presented here, semantic role labeling is usually divided into four steps: predicate identification and disambiguation, and argument identification and classification (Conia and Navigli, 2022). Current state-of-the-art semantic role labeling models build upon large pretrained language models such as BERT. In particular, the current best-performing model operating on German appears to be the multilingual one developed by Conia et al. (2021; see also Conia and Navigli, 2020).

Nevertheless, we have indications that much simpler models for semantic role labeling perform quite close to the state of the art. For instance, Conia and Navigli (2020) report that the monolingual BERT baseline model provided by Shi and Lin (2019) performs nearly equally as good as their more complex (multilingual) model on English. Essentially, the model by Shi and Lin performs argument identification by taking BERT's output representation and feeding it through a BiLSTM layer to predict BIO-encoded predicate labels.

However, they only fine-tune the BiLSTM layer; the attention weights of BERT remain fixed. Current research on Named Entity Recognition (Schweter and Akbik, 2021) and—closer to the speaker attribution task—recognition of speech, thought, and writing representation (Ehrmanntraut et al., 2023) suggests that rather than adding a BiLSTM layer, fine-tuning the Transformer's attention weights allows to predict the respective labels from the token encoding in the final Transformer layer alone. This Transformer-Linear variant corresponds to the now usual "BERT for token classification" setup and appears to be competitive, and often even outperforming model variants with BiLSTM decoders. My system directly follows this approach.

3 Base Model and Domain Adaptation

My system is based on the BERT Transformer model GBERT$_{\text{Large}}$ (i.e., deepset/gbert-large, Chan et al., 2020). Following Gururangan et al. (2020) and Konle and Jannidis (2020), I performed a domain adaptation of the model by continuing pre-training on a second, separate corpus of speeches. The corpus extends the SpkAtt training speeches with additional speeches held in the German Bundestag during the 9th–20th legislative period, from 1980 until April 2023 (757 MB). This results in the BERT model GePaBERT. The speeches were automatically prepared from the publicly available plenary protocols[5], using the extraction pipeline Open Discourse[6] (cf. Richter et al., 2023). Speeches that are present in the development or test split of the SpkAtt task were excluded, so that the predictive accuracy measured on the held-out development/test split actually re-

[4] https://github.com/aehrm/spkatt_gepade

[5] https://www.bundestag.de/services/opendata

[6] https://opendiscourse.de; https://github.com/open-discourse/open-discourse

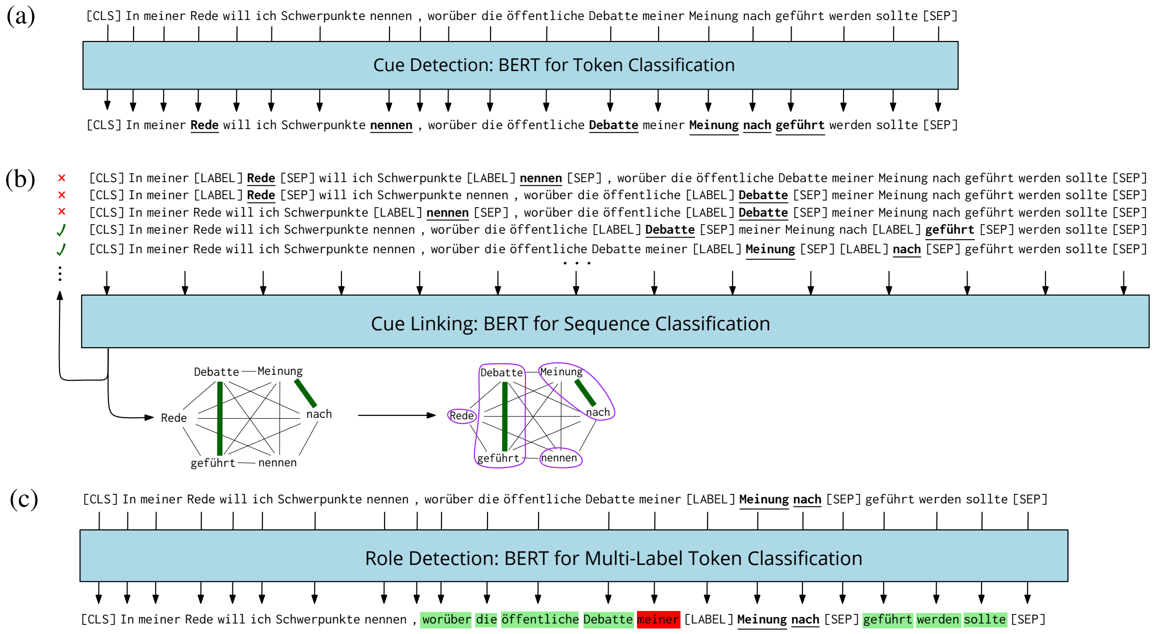

(a)
[CLS] In meiner Rede will ich Schwerpunkte nennen , worüber die öffentliche Debatte meiner Meinung nach geführt werden sollte [SEP]
Cue Detection: BERT for Token Classification
[CLS] In meiner Rede will ich Schwerpunkte nennen , worüber die öffentliche Debatte meiner Meinung nach geführt werden sollte [SEP]
(b)
✗ [CLS] In meiner [LABEL] Rede [SEP] will ich Schwerpunkte [LABEL] nennen [SEP] , worüber die öffentliche Debatte meiner Meinung nach geführt werden sollte [SEP]
✗ [CLS] In meiner [LABEL] Rede [SEP] will ich Schwerpunkte nennen , worüber die öffentliche [LABEL] Debatte [SEP] meiner Meinung nach geführt werden sollte [SEP]
✗ [CLS] In meiner Rede will ich Schwerpunkte [LABEL] nennen [SEP] , worüber die öffentliche [LABEL] Debatte [SEP] meiner Meinung nach geführt werden sollte [SEP]
✓ [CLS] In meiner Rede will ich Schwerpunkte nennen , worüber die öffentliche [LABEL] Debatte [SEP] meiner Meinung nach [LABEL] geführt [SEP] werden sollte [SEP]
✓ [CLS] In meiner Rede will ich Schwerpunkte nennen , worüber die öffentliche Debatte meiner [LABEL] Meinung [SEP] [LABEL] nach [SEP] geführt werden sollte [SEP]
Cue Linking: BERT for Sequence Classification
Debatte
Meinung
Rede
nach
geführt
nennen
Debatte
Meinung
Rede
nach
geführt
nennen
(c)
[CLS] In meiner Rede will ich Schwerpunkte nennen , worüber die öffentliche Debatte meiner [LABEL] Meinung nach [SEP] geführt werden sollte [SEP]
Role Detection: BERT for Multi-Label Token Classification
[CLS] In meiner Rede will ich Schwerpunkte nennen , worüber die öffentliche Debatte meiner [LABEL] Meinung nach [SEP] geführt werden sollte [SEP]

Figure 2: Overview of the system architecture. (a) The first component performs a token classification to detect cue words. (b) The second component performs a sequence classification to predict potential cue links on all pairs of cue words through contextualized cue-aware input sequences. (Not all such sequences are shown.) On the graph induced by the cue words resp. positive (green) links, the system picks the connected components (circled purple) as cue spans. (c) The third component performs a multi-label token classification to detect role words corresponding to the respective highlighted cue span.

flects the accuracy on data the system has never seen at all, e.g., speeches held after April 2023.

Training was done on 5 epochs, with a batch size of 8, and a learning rate of 2×10^{-5}, linearly decreasing to zero. (Training took approximately 140 GPU hours on two GTX 1080 TI GPUs, each with a device batch size of 2, and 2 gradient accumulation steps.) The final model GePaBERT is made available on the Huggingface hub.[7]

4 System Overview

My system splits the task into three components: (a) Detection of cue word, i.e. word that are covered by cue spans. (b) Joining individual cue word through the detection of cue links, in order to form cue spans. (c) For each cue span, given that specific cue span, infer the associated role spans. Figure 2 gives a sketch of the system. All three components are implemented by fine-tuning the above domain-adapted BERT model GePaBERT, respectively, employed in a token classification or sequence classification setup.

Instead of fully fine-tuning BERT models, the system builds upon LoRA adapters (Hu et al., 2021): rather than training all Transformer weights, the pre-trained weights are frozen, but trainable rank decomposition matrices are injected into each attention layer of the Transformer architecture. This reduces the number of trainable parameters and accelerates fine-tuning. To this end, the system is implemented through the PEFT library provided by the Huggingface API[8] (Mangrulkar et al., 2022; Wolf et al., 2020).

4.1 Cue Detection

The detection of cue words is achieved using a token classification by the first BERT model, fine-tuned for this task. Following standard practice, the model performs a token-level binary logistic regression, using BERT's output representation of the respective first wordpiece token of that particular word. Thus, the models differentiates between non-cue words and cue words. In this component and all the following, for each the regression weights and the respective Transformer's attention weights (through LoRA) are trained to minimize the binary cross entropy loss of the token classification against gold labels. Training was done over 30 epochs with a batch size of 4 and a learning rate of 5×10^{-5}. (In total, fine-tuning all three components took approx-

[7] https://huggingface.co/aehrm/gepabert

[8] https://github.com/huggingface/peft

imately 6 GPU hours on a single GTX 1080 TI GPU.)

The model performs this token classification for each sentence, but adds additional context by prepending the five preceding sentences, and appending the five following sentences to the sequence, both during training and inference. After inference, we obtain a set of predicted individual cue words.

4.2 Cue Linking

The second component joins individual cue words into cue spans. This is necessary since, even within the same sentence, there may be multiple different cue spans, each with their own associated role spans. (See, e.g., Figure 1(e) vs. (f).) To this end, the component predicts whether two cue words belong to the same span. Given two cue words, we first derive a cue-aware input sequence that highlights the two cue words. Then, we let the second BERT model perform a sequence classification on the input sequence, predicting whether the two highlighted cue words belong to the same span or to different spans. During inference, these link predictions are used to calculate a partition of cue words into spans.

In order to encode the two focused cue words into a cue-aware manner, a new special token `[LABEL]` was introduced, and the input sequence is designed as "`[CLS]` left context `[LABEL]` cue no. 1 `[SEP]` center context `[LABEL]` cue no. 2 `[SEP]` right context `[SEP]`". As usual, the sequence prediction is calculated using a binary logistic regression on BERT's output representation of the initial `[CLS]` token.

This classification is performed on all pairs of cue words that appear in the same sentence. (In fact, no cue span appears to span over multiple sentences.) The model is trained on all gold cue words, predicting whether two focused cue words are indeed contained in the same (gold) cue span. Again, the five preceding/following sentences were added to the left/right context.

During inference, the model takes the cue words predicted by the previous component. To now derive the actual partition into cue spans, set up a graph structure with every (predicted) cue word as vertex, and adding edges between two vertices if the classifier predicted a link between the two respective cue words. Finally, the model enumerates the connected components of that graph as prediction for the cue spans. While an enumeration of maximal cliques would also be an option—especially since under gold predictions, the connected components are always cliques—the component relaxes this condition and focuses only on connected components. In fact, no noticeable difference in performance between these two approaches could be observed.

4.3 Role Detection

The last component predicts the role spans, *given* a specific cue span, in the context surrounding the cue span. Like the first component, this component fine-tunes the third BERT model to perform a multi-label token classification, which, for each token, predicts to which role span(s) the respective token belongs. Note that a multi-label classification is needed since, even in the same speech event, a word may belong to *multiple* different role spans, even when associated with the same cue span. (Cf. Figure 1(c), which appears verbatim in the official GePaDe training dataset.)

This multi-label classification is modelled as seven independent binary classifiers (one for each role label *Source*, *Message*, *Topic*, . . . , i.e., binary relevance method). Again following standard practice, similar to the cue detection, each one of the classifiers is implemented as independent token-level binary logistic regression on (the same) output representation from BERT.

As input sequence, the model takes the sentence that contains the cue span, plus the five preceding and following sentences: for one, since role spans could also cover tokens from preceding/following sentences (cf. Figure 1(d)); for another, to give BERT more context to, e.g., disambiguate what the demonstrative pronoun *das* in Figure 1(b) actually refers to. The model encodes the sequence in a cue-aware manner similar to the previous component: again, the special token `[LABEL]` highlight contiguous tokens from the cue span. For instance, the speech event depicted in Figure 1(b) would be encoded as `[CLS] Frau Merkel , laut Medien [LABEL] nahm [SEP] die Bundesregierung das aber nicht [LABEL] zur Kenntnis [SEP] . [SEP]`.

This component has been trained on the gold annotation objects (i.e., given gold cue span, predict the gold role spans). During inference, the component takes the cue spans predicted by the previous component, and for each cue span, predicts the associated role spans, and finally returns complete annotation objects by combining the cue spans with the respective predicted role spans.

5 Results and Error Analysis

5.1 Metric

Since no code for the official SpkAtt scorer is available, we will, in the further course of the paper, instead resort to the following *matching-based precision/recall* as a guiding metric, which should be approximately in line with the informal descriptions given by the task organizers.

Consider one gold annotation A and one predicted annotation $\hat{A}$. For each of the eight shared task classes (*Cue*, *Source*, *Message*, etc.), calculate the recall between the gold span and the predicted span, each time on the token level.[9] Then, form the micro-averaged recall over all classes to get the *annotation recall* $R(A, \hat{A})$ between gold and predicted annotation.

Now, to calculate recall between *sets* of gold annotations and predictions, set up a complete bipartite graph between gold annotations and predictions. Weight each edge between gold A and predicted $\hat{A}$ according to $R(A, \hat{A})$. Then, determine a maximum-weight matching in that bipartite graph. The *matching-based recall* is the average of $R(A_i, \hat{A}_{i'})$, taken over all gold annotations A_i, where $\hat{A}_{i'}$ is the matched mate of A_i. (If A_i has no mate, then it contributes recall 0 to the average.)

Precision is computed in a symmetric fashion. Calculate micro-averaged annotation precision $P(A, \hat{A})$, and then calculate the maximum-weight matching with respect to a bipartite graph weighted by $P(A, \hat{A})$. The *matching-based precision* is the average over $P(A_{i'}, \hat{A}_i)$ taken over all predicted annotations $\hat{A}_i$, where $A_{i'}$ is the matched mate of $\hat{A}_i$. (Again, if $\hat{A}_i$ has not mate, it contributes precision 0 to the average.)

Now the matching-based F1 score Match-F1 is the harmonic mean between matching-based precision and recall. Note that the maximum-weight matchings calculated for precision resp. recall may not be identical.

	Full Task (1a)			**Gold cues given (1b)**		
	Match-Prec.	Rec.	F1	Prec.	Rec.	F1
Dev Set	84.3	85.2	84.8	92.7	92.1	92.4
only cues	90.8	92.2	91.5	—	—	—
only roles	81.0	83.1	82.0	87.9	89.3	88.6
	SpkAtt-Prec.	Rec.	F1	Prec.	Rec.	F1
Test Set	78.9	87.3	82.8	92.1	91.3	91.7
only cues	89.7	88.9	89.3	—	—	—
only roles	77.7	87.1	82.1	91.1	90.2	90.7

Table 1: Results of the system on the development set (evaluated using Match-Precision/Recall/F1) and on the test set (evaluated by the task organizers, using their SpkAtt-Precision/Recall/F1). "Only cues" resp. "only roles" refers to the metric variant where only cue spans resp. role spans are considered in the calculation. All scores are given in percentage points.

5.2 Quantitative Results

The organizers designed the task with two subsettings: In the full task (1a), predict cue spans together with corresponding roles. In the role labeling task (1b), the gold cue spans are given, and the task consists in predicting only the corresponding roles.

The proposed system was evaluated on two datasets: one, the provided development split of the GePaDe dataset. Second, on a blind test split, for which the gold annotations were only available to the shared task organizers. On both datasets, the system was evaluated with respect to both subtasks 1a and 1b. However, the metrics employed differ between the datasets: for the test set, the gold annotations are not publicly available, thus only the metrics returned by the task organizers are reported, denoted by SpkAtt-Precision/Recall/F1, who ran their closed-source official scorer on the submitted predictions.[10] In the development split, I used only the matching-based precision/recall as outlined above in Sec. 5.1, denoted with Match-Precision/Recall/F1.

Table 1 presents the results on the two datasets and the two task settings. Broadly, the results suggest that, even in this relatively simple setup, this BERT-based baseline already gives surprisingly steady performance. As we expect, this even increases in the second subtask (1b), where the gold cues triggering the speech events are given.

[9] I.e., when s is the gold span and s' is the predicted span of a particular class, then $|s \cap s'|$ is the number of true positives, $|s \setminus s'|$ is the number of false negatives, and $|s' \setminus s|$ is the number of false positives.

[10] The two respective predictions were submitted to CodaLab on July 30, 23:00 (No. 16) for task 1a and on August 2, 12:08 (No. 19) for task 1b. For task 1a, I thus report the performance of the second-last submission, not the last submission for task 1a (No. 17) which the task organizers intended to treat as the final official submission for task 1a. This final submission for task 1a differs to the one reported here only in the cue linking algorithm; the respective performances are nearly identical. (82.73 vs. 82.84 SpkAtt-F1 points for the system reported here.)

	Prediction with gold cues given (1b)			
Role Class	Match-Prec.	Rec.	F1	# train instances
Source	93.3	96.4	94.8	3337
Message	88.6	91.3	89.9	3242
Topic	70.8	83.1	76.4	871
Addressee	75.9	91.3	82.9	495
Particle	88.4	90.5	89.4	359
Medium	65.7	78.3	71.5	228
Evidence	77.8	69.9	73.6	80

Table 2: Breakdown on the system's performance on the development set in the role labeling task, when gold cues are given (1b), where metrics are calculated for each individual role class. The last column refers to the number of role spans per class present in the training split. All scores are given in percentage points.

Table 2 shows the performance for the role labeling task (1b) on the development set, broken down for each of the seven role classes. We can observe a clear trend that classes occurring less frequently in the training set are recognized less accurate. Additionally, a more detailed quantitative analysis (not shown here) indicates that the system slightly struggles to differentiate between *Topic* vs. *Message*, and *Medium* vs. *Message*.

To further assess the impact of domain adaptation of the chosen base BERT model and the variability introduced by the random fine-tuning, I repeated the fine-tuning five times on GePaBERT, but also on GBERT$_{\text{Large}}$ (i.e., GePaBERT before domain adaptation), and GBERT$_{\text{Base}}$ (i.e., the smaller variant `deepset/gbert-base` with fewer layers). Note that this was only conducted after the shared task's system submission deadline. Table 3 reports the measured accuracies, given in empirical mean and standard deviation.

As we expect, we clearly observe a jump in performance between the "base-size" and "large-size" variant of BERT. However, the domain adaptation of GBERT$_{\text{Large}}$ to GePaBERT, as outlined in Section 3, appears to have only minimal or no effect at all. I do not have a good explanation for this behavior. For one, maybe more data is necessary for an effective domain adaptation; for another, perhaps further hyperparameter studies for the domain adaptation are necessary to find the optimal pre-training procedure. Along this, pre-training itself should also be extended beyond the current five epochs, something for which there was insufficient time during the development of the system. Or, arguing in the other direction, the observed performances by both the GePaBERT and GBERT$_{\text{Large}}$ might suggest that both models already hit the same performance ceiling, which might be much harder to break through.

While these results contradict the findings of Konle and Jannidis (2020)—who were able to achieve substantial improvements using domain adaptation—it should be noted that they also included the test set of the corresponding downstream task during the pre-training of the base language model (though not during the fine-tuning). As explained in Section 3, this was not done for the system at hand, in order to measure accuracy against future data that the model has never seen. Yet, as Konle and Jannidis hypothesize, precisely this pre-training on the (unlabeled) test data may allow the language model to build a better representation of the test data, helping in solving the downstream task. Nevertheless, such an increase in accuracy comes with the disadvantage that, when the system is applied on new unlabeled data, the entire base language model may possibly need to be pre-trained again on this new data to maintain the same performance. In total, further research towards domain adaptation (especially in Computational Humanities resp. Computational Social Sciences) is needed.

5.3 Qualitative Error Analysis

Next to the quantitative analysis of the system's performance, I also performed a manual error analysis of the system's predictions on the development set. Concerning the cues, it appears that the system is particularly struggling with recognizing nominal triggers, e.g., "*Als nächster Redner hat das Wort [...],*" "*Wo waren Sie bei den Koalitionsverhandlungen?,*" "*Die richtige Antwort bei Betrug, [...]*" have not been predicted as cues, whereas the system erroneously predicts, e.g, "*Die Ziele des Gesetzentwurfs sind nicht einmal falsch, [...],*" "*An dieser Einsicht hat sich [...] nichts verändert,*" etc. Furthermore, in many of the false-positive cases, the presence of speech, thought, or writing representation, is ambiguous, e.g., in "*Die Mehrzahl der Handwerksbetriebe beurteilt [...] die wirtschaftliche Lage als sehr gut"* the verb is predicted as cue, but not annotated as such.

Concerning the role prediction, I am focusing on the results for the task setting where gold cues are given (1b). The manual analysis confirms the observation already outlined above, that the system is struggling to differentiate between *Medium*, *Topic*,

	Full Task (1a)			Gold cues given (1b)	
Model	**Match-F1**	**(on cues only)**	**(on roles only)**	**Match-F1**	**(on roles only)**
GBERT$_{\text{Base}}$	80.09 ± 1.12	90.06 ± 0.54	75.23 ± 2.60	88.66 ± 1.44	82.60 ± 3.09
GBERT$_{\text{Large}}$	**84.16 ± 0.98**	90.84 ± 0.78	**81.76 ± 1.02**	**92.07 ± 0.60**	**88.07 ± 0.90**
GePaBERT	84.12 ± 0.74	**91.36 ± 0.44**	81.24 ± 1.07	91.55 ± 0.75	87.18 ± 1.13

Table 3: F1-Scores on five fine-tuning runs, evaluated on the development set, presented as empirical mean and standard deviation. All scores are percentage points. Highest score for each column is highlighted bold.

and *Evidence*. In fact, the annotation guidelines intensively elaborate on a differentiation between these classes, which could hint at an inherent complexity of this task.

The second major source of errors seems to be that certain phrases are not recognized as roles at all by the system. In particular, there appears to be a disagreement between the system and the gold annotation as to which phrases belong to the *Message* and which do not. For instance, in the gold annotation "*Ich sage Ihnen eines, Herr Mützenich – das sage ich auch den Kollegen von Grünen und Linkspartei –: Wir diskutieren gerne über [Vermögenssteuern]. Jetzt müssen wir uns nur darum kümmern, dass es überhaupt noch eine wirtschaftliche Substanz gibt [...].*" the second and third sentence is part of the gold message span, but not in the prediction. Symmetric, in the prediction "*Fast alle mit Kindern unter drei Jahren arbeiten in Teilzeit, und – das sage ich ganz offen – es ist zu befürchten, dass sie aufgrund geringer Gehälter jetzt beruflich zurückstecken.*" the phrase after the parenthesis is not part of the gold role.

5.4 Testing Political Bias

As last part of my analysis, I want to provide some explorations on potential biases of my system along a political axis. The system might be used in more downstream tasks inferring information from German Bundestag debates, e.g., in a quantitative analysis comparing the speeches of the different parliamentary groups. Thus, to allow neutral inferences on such textual datasets, it is imperative to investigate potential imbalances in system performance, in particular between parliamentary groups, in order to avoid any unintended biases towards or against certain parliamentary groups.

For this, I am focusing on the system's accuracy, comparing the accuracies on the development set along the different parliamentary groups. In the following, I am referring with parliamentary groups to the groups (*Fraktionen*) that were represented in the 19th and 20th Bundestag. The development set speeches were pooled according to the parliamentary group the respective speaker is member of, as indicated by the GePaDe dataset.

To now infer differences in F1 score between the parliamentary groups, I performed parameter estimations through two separate regressions on the Match-Precision resp. Match-Recall on the development set. Here, the observations are the micro-averaged precisions resp. recalls on the speech events, that are used to compute Match-Precision resp. Match-recall. Since the distribution of the individual observations is highly bimodal (e.g., for each predicted speech event, micro-precision is either around 100 % or around 0 %) I chose to perform a Bayesian hierarchical beta-binomial regression. For instance, in the Match-Precision regression, for each observation the predicted-positive count is the number of trials, and the true-positive count is the number of successes. Now, instead of assuming a fixed success probability, the success probability is rather sampled, individually for each observation, from a high-level beta distribution corresponding to the respective parliamentary group. We are interested in inferring the shape parameters of these beta distributions. I particularly allowed in the prior for U-shaped beta distributions. Inference was conducted with PyMC[11] and the provided MCMC sampler.

The Bayesian models allow us to sample the mean parameter from the precision resp. recall beta distribution, and by taking the harmonic mean, we can visualize the posterior distributions of the Match-F1 score, for each parliamentary group respectively, as in Figure 3(a). Visually, we see how the estimates for F1 scores vary for each of the respective parliamentary group. The effect is most prominent between the SPD and LINKE group, where the model estimates the mean of F1 score for the particular group at 80.9 vs. 86.7 percent points. (Pr = 0.88 for a difference of > 5 percent points between the two groups.) However, it needs to be further examined whether this difference actually

[11] https://www.pymc.io

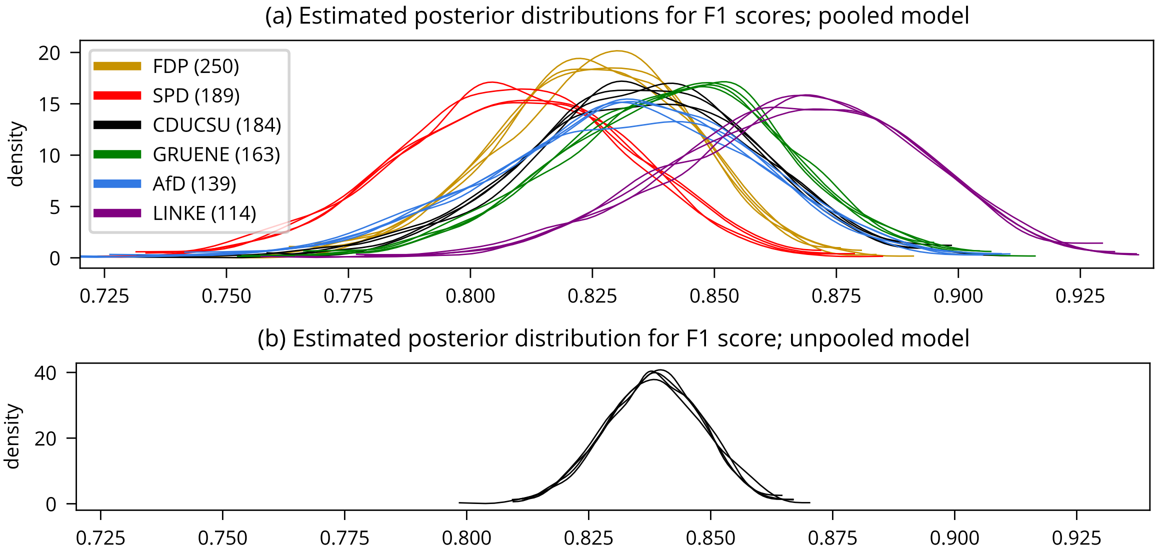


Figure 3: Estimated posterior distributions for the Match-F1 score on the development set. Each line corresponds to one of the four chains of the MCMC posterior draw. (a) Pooled model, where each parliamentary group has their own parameters and F1 posterior. The numbers in the legend indicate the number of observations per parliamentary group. (b) As a comparison, posterior predicted from the unpooled model, where every parliamentary group shares the same parameters, and thus F1 score is distributed identically over all groups.

reflects a certain bias of the system towards certain textual phenomena or speech content, or whether this may be an effect of the random development split, where, e.g., the speeches of the LINKE group only randomly happen to be 'easy' ones.

Even from a statistical point of view, we should not overestimate this result. Especially in light of the low number of observations in the development split, we might possibly see the result of the model over-fitting the data, thus erroneously moving the F1 distributions apart. In fact, we can compare the previous pooled model with a unpooled model, where the distributions of the political groups are the same (Figure 3(b)). An estimation of their respective expected log pointwise predictive density shows that these are largely equal (-1097.0 ± 41.7 for the pooled model vs. -1084.0 ± 41.6 for the unpooled one, where higher is better), ranking no model clearly above the other (cf. Vehtari et al., 2017).

In total, we see some indication of a difference in system performance between the parliamentary groups, at least in the development dataset. Nonetheless, further investigations are required to verify if these imbalances remain stable even when moving to larger test sets. For this particular case at least, a model comparison indicates no significant statistical evidence of a performance imbalance along different parliamentary groups.

6 Conclusion

The present paper summarizes my submission for the Shared Task on Speaker Attribution SpkAtt-2023, specifically task 1 for attribution in parliamentary speeches. The system handles this task as a collection of token classification resp. sequence classification tasks, using BERT as base language model. Thus, the present system offers a simple BERT-based baseline model, which, despite its minimal architecture, provides a steady baseline. Even the variant based on the smaller GBERT$_{\text{Base}}$ model appears to have minimal performance losses, making it applicable to settings with less compute resources. In contrast, a domain adaptation through continued fine-tuning on a corpus of speeches from the German Bundestag led to no significant improvement. The error analysis indicates that the system is mostly struggling primarily with ambiguous 'edge cases,' where it appears to be not even entirely clear what the correct annotation would be. A quantitative comparison of the system's performance across the different parliamentary groups shows no strong evidence towards a potential imbalance. Overall, these results indicate the applicability of the system in further downstream analyses, e.g., in quantitative discourse studies of parliamentary debates.

References

Branden Chan, Stefan Schweter, and Timo Möller. 2020. German's next language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Simone Conia, Andrea Bacciu, and Roberto Navigli. 2021. Unifying cross-lingual semantic role labeling with heterogeneous linguistic resources. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 338–351, Online. Association for Computational Linguistics.

Simone Conia and Roberto Navigli. 2020. Bridging the gap in multilingual semantic role labeling: a language-agnostic approach. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1396–1410, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Simone Conia and Roberto Navigli. 2022. Probing for predicate argument structures in pretrained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4622–4632, Dublin, Ireland. Association for Computational Linguistics.

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

Anton Ehrmanntraut, Leonard Konle, and Fotis Jannidis. 2023. LLpro: A literary language processing pipeline for German narrative texts. In *Proceedings of the 19th Conference on Natural Language Processing (KONVENS 2023)*, Ingolstadt, Germany. KONVENS 2023 Organizers. To be published.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-rank adaptation of large language models. ArXiv: 2106.09685.

Leonard Konle and Fotis Jannidis. 2020. Domain and task adaptive pretraining for language models. In *Proceedings of the Workshop on Computational Humanities Research (CHR 2020)*, volume 2723 of *CEUR Workshop Proceedings*, pages 248–256, Amsterdam, the Netherlands.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, and Sayak Paul. 2022. PEFT: State-of-the-art parameter-efficient fine-tuning methods. GitHub Repository.

Ines Rehbein, Fynn Petersen-Frey, Annelen Brunner, Josef Ruppenhofer, Chris Biemann, and Simone Paolo Ponzetto. 2023. Overview of the GermEval 2023 Shared Task on Speaker Attribution in Newswire and Parliamentary Debates. In *The GermEval 2023 Shared Task at KONVENS 2023*, Ingolstadt, Germany.

Florian Richter, Philipp Koch, Oliver Franke, Jakob Kraus, Lukas Warode, Fabrizio Kuruc, Stella Heine, and Konstantin Schöps. 2023. Open Discourse: Towards the first fully comprehensive and annotated corpus of the parliamentary protocols of the german bundestag. SocArXiv: dx87u.

Stefan Schweter and Alan Akbik. 2021. FLERT: Document-level features for named entity recognition. ArXiv: 2011.06993.

Peng Shi and Jimmy Lin. 2019. Simple BERT models for relation extraction and semantic role labeling. ArXiv: 1904.05255.

Aki Vehtari, Andrew Gelman, and Jonah Gabry. 2017. Practical bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27(5):1413–1432.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Index of Authors