

# GerPS-Compare: Comparing NER methods for legal norm analysis

Sarah T. Bachinger<sup>1,2</sup>, Christoph Unger<sup>3</sup>, Robin Erd<sup>1,2</sup>, Leila Feddoul<sup>1,2</sup>,  
Clara Lachenmaier<sup>3</sup>, Sina Zarriß<sup>3</sup>, Birgitta König-Ries<sup>1</sup>,

<sup>1</sup>Heinz Nixdorf Chair for Distributed Information Systems, Friedrich Schiller University Jena, Germany

<sup>2</sup>Competence Center Digital Research (zedif), Friedrich Schiller University Jena, Germany

<sup>3</sup>Computational Linguistics, Department of Linguistics, Bielefeld University, Germany

Correspondence: [sarah.bachinger@uni-jena.de](mailto:sarah.bachinger@uni-jena.de)

## Abstract

We apply NER to a particular sub-genre of legal texts in German: the genre of legal norms regulating administrative processes in public service administration. The analysis of such texts involves identifying stretches of text that instantiate one of ten classes identified by public service administration professionals. We investigate and compare three methods for performing Named Entity Recognition (NER) to detect these classes: a Rule-based system, deep discriminative models, and a deep generative model. Our results show that Deep Discriminative models outperform both the Rule-based system as well as the Deep Generative model, the latter two roughly performing equally well, outperforming each other in different classes. The main cause for this somewhat surprising result is arguably the fact that the classes used in the analysis are semantically and syntactically heterogeneous, in contrast to the classes used in more standard NER tasks. Deep Discriminative models appear to be better equipped for dealing with this heterogeneity than both generic LLMs and human linguists designing rule-based NER systems.

## 1 Introduction

The application of Natural Language Processing (NLP) to legal texts in German is a relatively new development, starting in an era where deep discriminative approaches to Named Entity Recognition (NER) have already been established as state of the art technologies. Hence, many implementations of NER for legal documents in German turn to deep discriminative ML approaches directly, without systematically comparing these technologies to alternative approaches (Leitner et al., 2019; Darji et al., 2023; Peikert et al., 2022)

We aim to fill this gap by comparing three different approaches: rule-based methods (symbolic AI), deep discriminative models and deep generative models. Moreover, we run this comparison on

a dataset that is very close to real-world applications that NER in the legal domain may be used for, rather than re-using the standard NER benchmark datasets. The application scenario for the GerPS-NER dataset that we chose to work with (Feddoul et al., 2024) is to assist humans in analyzing legal bases<sup>1</sup> with the goal of creating digitized versions of administrative process schemata in the public administration.<sup>2</sup> Since the dataset we chose includes some highly structured sub-types of legal language, we believe that it makes sense to include a rule-based approach in the comparison, as this approach is known to be well suited for such texts.

Each of the approaches we compare comes with different trade-offs in terms of development effort and adaptability, amounts of training data needed and prediction accuracy. While rule-based approaches can deal well with structured text, it is a time-consuming task to create the rulesets, they are relatively sensitive to errors in the dataset and can only detect known patterns. Deep generative systems bring with them a lot of contextual knowledge about the world that the data is embedded in. On the other hand, they require large amounts of computational power, and they are expecting continuous text as input and output, making them more difficult to work with when strict formats have to be adhered to. Deep discriminative models have long been used in NLP tasks due to the relatively reliable performance they deliver. The challenge with deep discriminative models is that they require large amounts of training data.

The remainder of this paper is organized as follows: [Section 2](#) outlines the relevant literature. [Section 3](#) describes the methodology used in the study.

<sup>1</sup>In Germany, the Federal Information Management <https://fportal.de/glossar> provides standardized methods for analyzing such legal bases.

<sup>2</sup>Eventually, it is planned to integrate one or multiple approaches into a software for use by interested public administrations, so practical considerations regarding the approaches must also be kept in mind.

Section 4 describes the details of the model implementations. Section 5 presents the results, and Section 6 discusses their implications. Finally, Section 7 concludes the paper and suggests areas for future research.

## 2 Related work

### 2.1 NER in general

General surveys of approaches to NER can be found in [Yadav and Bethard \(2019\)](#) and [Pakhale \(2023\)](#). Both surveys discuss deep discriminative approaches, which have become state of the art in NER tasks until recently, when large language model-based approaches (in the following deep generative approach) appeared on the scene ([Wang et al., 2023](#); [Bogdanov et al., 2024](#); [Ye et al., 2024](#); [Monajatipoor et al., 2024](#); [Jung et al., 2024](#); [Zhang et al., 2024](#); [Naguib et al., 2024](#)). Different authors compare the effectiveness of LLM-based NER in the legal domain: for LegalLens, [Bernsohn et al. \(2024\)](#) compared BERT models with open source LLMs on the domain of legal violation identification. [Joshi et al. \(2024\)](#) proposed “IL-TUR, a benchmark for Indian Legal Text Understanding and Reasoning” and offer among other things a LLM-based pipeline for the benchmark. With LAiW, [Dai et al. \(2023\)](#) propose a benchmark for Chinese legal LLMs. [Bachinger et al. \(2024\)](#) systematically evaluate different open source LLMs for their effectiveness in German text generation and use Prompt Engineering and Fewshot Prompting for NER on German legal texts. Their investigation on a small subset show optimistic results for one of their prompting schemes in combination with the German open source LLM LeoLM.

While the success of the deep discriminative, and to a lesser degree the deep generative, approaches to NER seems to have made rule-based approaches obsolete, the simplicity and robustness of rule-based approaches make them still strong competitors at least in some domains. For instance, [Gorinski et al. \(2019\)](#) systematically compare a rule-based NER system for electronic health records with deep learning and transfer learning systems. They found that the hand crafted rule-based system consistently outperforms both the transfer learning and the deep learning systems, reaching an overall F1-score of 0.95.

In systematically comparing rule-based approaches not only to deep discriminative, but also to deep generative approaches we hope to provide

a broader evaluations of the options currently available for NER applications in the legal domain.

### 2.2 NER in legal documents

NER systems for legal documents have been developed for a long time ([Dozier et al., 2010](#)). Rule-based approaches to NER in legal documents are mostly developed for languages lacking robust resources for ML development, such as Afan Oromo ([Raja et al., 2019](#)) or Arabic ([Abdallah et al., 2012](#)). The latter work stands out by combining a rule-based approach with machine learning and evaluating the effectiveness of both approaches. The authors find that the combined approach improves the F1-score by 8 – 14% compared to either the rule-based approach or the machine learning approach alone.

The entities recognized by NER systems for legal documents usually center around entities related to the court system (*judge, lawyer, court, court decision, jurisdiction, etc.*) and references to sections in law texts. Our work seeks to find entities related to legal norms for the administration of public services.

### 2.3 NER in German legal documents

NER systems for legal documents in German are mostly based on deep discriminative approaches. Thus [Leitner et al. \(2020\)](#) present a relatively large newly created dataset for German legal NER (German LER dataset) and also evaluate the performance of multiple differently configured BiLSTM-CRF models on this dataset. [Darji et al. \(2023\)](#) fine-tuned a German BERT model on the German LER dataset and present their results that are better than the results originally achieved by [Leitner et al. \(2020\)](#) when presenting the dataset. [Zöllner et al. \(2021\)](#) used, among others, the German LER dataset when they compared the effect of different pre-training techniques for small BERT models and presented modified fine-tuning processes which resulted in performance improvements. [Erd et al. \(2022\)](#) used the same two architectures that will also be used in this paper (BiLSTM-CRF, XLM-RoBERTa ([Conneau et al., 2020](#))) to evaluate and compare the performance improvements that different data augmentation methods and their combinations might achieve for NER tasks in the German legal domain. [Feddouli et al. \(2024\)](#) present GerPS-NER, a new corpus for NER on German legal texts covering the sub-genre of legal norms regulating the administration of public services.

In GerPS-NER, ten classes relevant for the analysis of this particular sub-genre are defined, which are intended to be used in aiding the digitization of public administration. While the classes used by Leitner et al. (2020) resemble more common Named Entities, the GerPS-NER corpus also includes more abstract concepts, such as *Bedingung* ‘condition’ (see Appendix A). Appendix C illustrates how these classes are brought to bear on the analysis of legal norms.

### 3 Concept

The workflow for our experiments is shown in Figure 1.

We use three different approaches to annotate German legal texts for the occurrence of expressions belonging to one of ten classes, as seen in Figure 1. The classes were previously derived by GerPS-NER (Feddouli et al., 2024).

The first approach is a **rule-based approach** with a linguist drafting suitable rules from gold standard examples.

To implement the **deep discriminative** approach, we selected two popular models that have also been used by Erd et al. (2022). The first is a BiLSTM-CRF model implemented with the FLAIR framework (Akbik et al., 2019), the second is the XLM-RoBERTa (XLM-R) (Conneau et al., 2020) transformer model, implemented using the FLERT extension (Schweter and Akbik, 2021) of the framework.

The **deep generative** approach is based on Bachinger et al. (2024) who compared several LLMs (Large Language Models) for their performance on the task of legal norm analysis on a small dataset. We use the prompting scheme they deemed the best consisting of the task description, three examples per class and the annotation guideline. We also use the German LLM LeoLM (Plüster, 2023) for prompting to see whether the promising results for the small dataset hold up in a systematic evaluation.

The **dataset** used in the evaluation is GerPS-NER published by Feddouli et al. (2024). We adapted it slightly for our purposes as we found tokenization problems in the corpus. We split the corpus in 20% development, 20% test and 60% training data. As shown in Figure 1, the train data split was only used by the deep discriminative approach, while the others used the dev split for creating rules and testing the code.

## 4 Implementation

In the following, we describe implementation details for the approaches. The code is available on zenodo (Anonymous, 2024).

### 4.1 Metrics for evaluation

As it is custom in the evaluation of NER tasks, we use the F1-score and the associated precision and recall values. For the overall evaluation we use the macro F1-score since in our application-scenario all classes are of equal importance. However, our main focus is on the per-class scores.

Measuring precision and recall, and hence calculating the F1-score, is essentially a token-based procedure. However, most of the entities we try to find cover spans of several tokens (see Appendix C). This opens up the possibility that a prediction based on a certain rule may not cover exactly the same span of tokens as the ground truth. Token-based evaluation metrics would systematically count spurious false positives and false negatives and lead to lower values of precision, recall and F1-score values when prediction and ground truth only partially overlap. For these reasons we have decided to supplement these token-based measures with span-based ones.

The span-based measure we propose to use is the intersection over union measure, or Jaccard similarity score. This metric is commonly used in image processing tasks, but Soleimani et al. (2021) uses it for measuring text-span overlap (text-span similarity). We calculate the Jaccard score for each class individually. Moreover, two of our approaches, the rule-based approach and the deep generative approach, split the corpus in a large number of small files containing one sentence each. This means that we must aggregate the results for each class per file and find a basis for a global evaluation. To do this, we determine the arithmetic mean of the Jaccard score values of all file inputs. In addition, we collect the number of inputs with Jaccard score 0, i.e. cases of zero-overlap between prediction and ground truth, and the number of total sentences where a given entity occurs. This information helps to interpret the arithmetic mean of Jaccard scores. After all, relative low values of the mean of the Jaccard scores could be due to two different scenarios: first, there might be a certain number of sentences with a high Jaccard score (near perfect overlap between prediction and ground truth), and second, all sentences may show some overlap between predic-

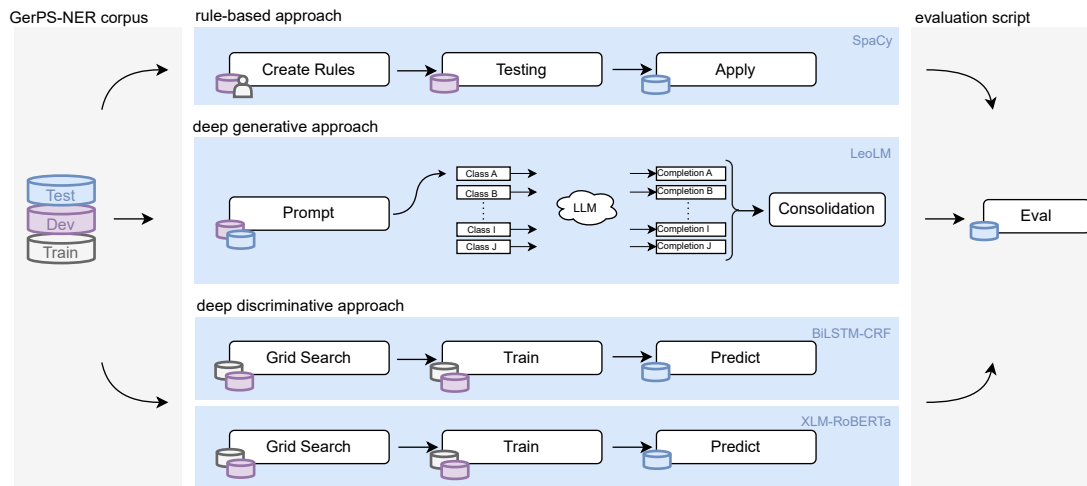


Figure 1: Overview of our workflow for comparing multiple machine learning approaches

tion and ground truth, but not a lot. Knowing the number of zero-overlap instances therefore helps to shed light on the interpretation of the mean value. Furthermore, we can calculate the proportion of zero-overlap instances to the total number of instances by dividing the former number with the latter, yielding a number between 0 (best case) and 1 (worst case), which furthermore sharpens the evaluation. Finally, we determine the median of the Jaccard scores. Thus, we base our evaluation of the respective approaches on a variety of metrics that need to be carefully interpreted.

## 4.2 Rule-based approach

The **rulebased approach** is implemented in the SpaCy framework, using token patterns and phrase patterns within SpaCy’s `EntityRuler`. Token patterns allow the use of morphological features in the rules. Phrase patterns match against exact word or phrase matches and are well suited to implement word-list based patterns (gazetteers). Several phrase patterns are dynamically constructed programmatically by looping over the tokens of the input text, applying filter functions defined in Python.

The design choice to utilize SpaCy’s `EntityRuler` helps to keep the rules simple and easily maintainable. However, it also means that it is not possible to use syntactic information such as provided by SpaCy’s dependency parser. This in turn means that some patterns, for instance those defining the entity *Bedingung* ‘condition’, are inherently limited in the depths of recursion they can cover.

Patterns for the entities *Handlungsgrundlage* ‘legal or formal grounds for the action described’ and *Hauptakteur* ‘main actor’ utilise word-list patterns (gazetteers) derived from the language model based on legal texts in German contained in the FLAIR framework `flair/ner-german-legal`. These patterns are augmented by patterns derived from the development set.

## 4.3 Deep discriminative approach

For the BiLSTM-CRF model, we use stacked German fastText (Bojanowski et al., 2017) and German forward and backward FLAIR embeddings (Akbik et al., 2018). The model is trained using the default Stochastic Gradient Descent without momentum, with gradients clipped at 5. The maximum number of training epochs is set to 150, but the learning rate is annealed based on performance on the development set, with training stopping early if the learning rate drops below 0.0001. Variational dropout is applied.

For the XLM-R model there are fewer parameters to configure. We chose this transformer model because preliminary studies on an early version of a subset of the corpus showed that it outperforms other German models we tested. The default setting in FLERT is to use the AdamW optimizer. We fine-tune the model with a learning rate that increases from 0 to 5 e-6 during the warm-up phase and then decreases linearly to 0 by the end of the training.

For both models, we conducted a grid search to select the hyperparameters. For the BiLSTM-CRF model, we found that a learning rate of 0.05 (from

the options of 0.05, 0.1, 0.2) and a batch size of 16 (from the options of 8, 16, 32) produced the best results on the development set. For the XLM-R transformer model, 30 fine-tuning epochs (from the options of 15, 20, 30) combined with a batch size of 1 (from the options of 1, 4) yielded the best performance.

#### 4.4 Deep generative approach

The approach works as follows: for every sentence from the dataset, ten prompts (one for each class) are created. These prompts contain the class definition in addition to the components mentioned above. The prompts are given to LeoLM and the completion is saved to a text file. The completions are checked for their length and for the content of the predictions, so that only predicted sentences containing the same tokens as the input sentence are processed further. Next, the valid predictions are consolidated into one sentence. Because there may be multiple predictions for a given token, [Bachinger et al. \(2024\)](#) use two different variants of sentence consolidation, a so-called optimistic and a pessimistic sentence consolidation. Optimistic sentences consolidation means that if one of the model’s possible predictions matches the ground truth, this particular prediction is chosen. Pessimistic sentence consolidation means that if there are multiple predictions for a token, a new class X is assigned that represents conflicting annotations. For each file in the test, we generate an optimistic (GenAI opt), a pessimistic (GenAI pes), and a gold standard IOB file from the dataset as the tokenization in this approach varies slightly from the dataset.

## 5 Results

This section presents the results of the model evaluation, divided into two parts: an overview of overall performance and a detailed analysis of individual model performances.

### 5.1 Summary of key findings

In this section, we outline our key findings. The micro F1-scores for the model predictions are presented in [Table 1](#), while the Jaccard score are shown in [Table 2](#). Additionally, these results are visualized in [Figure 2](#). Overall, the XLM-R model outperformed all other models across most classes, except for the *Datenfeld* ‘data field’ class. Contrary to expectations, the deep generative model

did not outperform the other models. Even the optimistic interpretation of its outputs resulted in the second-lowest macro F1-score, just above the pessimistic interpretation. The rule-based approach was the only one to surpass the XLM-R model performance in at least one class and also outperformed the deep generative approach in terms of overall macro F1-score.

### 5.2 Detailed performance analysis

The following two sections will take a closer look at the individual model performances.

**F1-score** Analyzing the results for the rule-based approach, the F1-scores for the respective classes are generally not very high, but range from 0.43 to 0.63. This indicates that there is not a lot of variation in the performance of the rules implementing the various classes. A similar trend is observed for the deep discriminative models, which perform consistently across all classes with a slightly broader and higher range, from 0.52 to 0.84. The deep generative approach provides two evaluation reports (see [Subsection 4.4](#)), with the F1-scores for the optimistic sentence consolidation exceeding those of the pessimistic one, as expected. Notably, the *Handlungsgrundlage* ‘legal grounds for action’ class is less affected by the pessimistic consolidation scheme and remains the best-performing class by a significant margin. The *Datenfeld* ‘data field’ class is an exception for all mentioned models, with scores as low as 0.07, 0.17, 0.03, 0.11 and 0.18 for BiLSTM-CRF, XLM-R, GenAI opt, GenAI pes and the rule-based approach, respectively, with the rule-based approach achieving the highest score. The XLM-R model achieved twice the score of the BiLSTM-CRF for the *Datenfeld* ‘data field’ class, despite both models generally yielding similar results. The *Datenfeld* ‘data field’ class has proven to be notoriously difficult to define, annotate manually and capture in rules. It should therefore be considered an outlier and may best be excluded from consideration.

**Jaccard** The Jaccard similarity score for a given class provides a measure of how closely the text spans marked as instantiating the class overlap between prediction and gold standard, in a given document. Since the corpus is split into many documents covering a sentence each, we take a score for every sentence and must look at the arithmetic mean value of these scores in order to understand how well the system’s prediction for a given class performs. However, the mean of the Jaccard simi-

Class	BiLSTM-CRF	XLM-R	LeoLM (opt)	LeoLM (pes)	Rule-based
Action	0.7443	<b>0.7621</b>	0.6102	0.0421	0.6049
Condition	0.8244	<b>0.8329</b>	0.4678	0.2240	0.5944
Data field	0.0721	0.1676	0.1076	0.0338	<b>0.1829</b>
Document	0.7661	<b>0.8126</b>	0.6144	0.0178	0.5861
Recipient of service	0.7674	<b>0.8004</b>	0.6828	0.0220	0.5531
Deadline	0.5967	<b>0.6569</b>	0.4699	0.1485	0.4813
Legal grounds for action	0.7985	<b>0.8362</b>	0.6643	0.4794	0.4450
Main actor	0.7315	<b>0.7724</b>	0.4239	0.0129	0.5747
Contributor	0.5276	<b>0.6173</b>	0.5020	0.1227	0.4258
Signaling word	0.8352	<b>0.8423</b>	0.3940	0.0701	0.6341
Macro F1-score	0.6058	<b>0.6455</b>	0.4488	0.1067	0.5082

Table 1: F1-Scores for the evaluated approaches by class and model. The best score for each class is highlighted in bold.

Class	BiLSTM-CRF		XLM-R		LeoLM (opt)		LeoLM (pes)		Rule-based	
	mean $\uparrow$	ratio $\downarrow$	mean $\uparrow$	ratio $\downarrow$	mean $\uparrow$	ratio $\downarrow$	mean $\uparrow$	ratio $\downarrow$	mean $\uparrow$	ratio $\downarrow$
Action	0.65	0.17	<b>0.67</b>	0.18	0.52	0.37	0.03	0.94	0.40	0.35
Contributor	0.32	0.60	<b>0.42</b>	0.51	0.27	0.63	0.04	0.90	0.18	0.74
Main actor	0.59	0.31	<b>0.65</b>	0.29	0.29	0.63	0.01	0.99	0.34	0.59
Recipient of service	0.59	0.32	<b>0.64</b>	0.29	0.56	0.38	0.02	0.98	0.33	0.58
Deadline	0.43	0.47	<b>0.49</b>	0.38	0.30	0.50	0.06	0.83	0.29	0.56
Condition	0.64	0.29	<b>0.67</b>	0.26	0.19	0.54	0.07	0.70	0.33	0.51
Document	0.62	0.29	<b>0.69</b>	0.22	0.52	0.43	0.02	0.98	0.39	0.50
Data field	0.14	0.77	<b>0.21</b>	0.71	0.03	0.95	0.01	0.98	0.07	0.85
Signaling word	0.74	0.19	<b>0.74</b>	0.19	0.29	0.63	0.05	0.93	0.44	0.49
Legal grounds for action	0.72	0.21	<b>0.74</b>	0.20	0.47	0.35	0.25	0.52	0.26	0.35

Table 2: Jaccard means for the evaluated approaches by class and model. The best score for each class is highlighted in bold.

larity scores needs to be interpreted in context of the zero-overlap to total count ratio and the median value, as discussed in Subsection 4.1. These values together provide another perspective on the model performance. Overall, the mean Jaccard scores generally reflect the performance distribution across classes observed with the F1-score, although they do give a somewhat different insight into the performance of individual classes. We discuss an example of the insights one can gain from a close analysis of the Jaccard score in the context of the Rule-based system in Subsection 6.1. One thing to note is that, while the F1-scores and Jaccard means differ between the XLM-R and BiLSTM-CRF models, the ratios for the *Datenfeld* ‘data field’ class are quite similar, at 0.77 and 0.71.

## 6 Discussion

### 6.1 Rule-based approach

The token-based F-score evaluation requires little comment. We therefore focus here on the span-based evaluation based on the Jaccard similarity score (or Intersection-over-Union measure). In order to see the value of adding the Jaccard score

analysis to the evaluation, let us consider the performance of the classes *Signalwort* ‘signaling word’ *Handlungsgrundlage* ‘legal grounds for action’ and *Bedingung* ‘condition’. We leave it to the reader to apply similar considerations to the data from other classes in different approaches as listed in Table 2.

Let us first consider the class *Signalwort* ‘signaling word’. This is the class with the highest Jaccard mean value, a value of 0.43. The ratio of complete prediction failures (zero-overlap cases) to the total number of occurrences of the class at 428 : 880 is 0.49. In other words, there is a medium high number of zero-overlap cases. This in turn means that the implementation misses a significant number of instances of the class, but when it does find an instance, the span it marks as belonging to the class overlaps significantly with the gold standard. This conclusion is reinforced by a relatively low median value of 0.32.

Consider now the class *Handlungsgrundlage* ‘legal grounds for action’. This class as the second lowest ratio of zero-overlap cases (216) to total counts (622), at a value of 0.35. The Jaccard score mean is 0.26 (0.258). This means that the imple-

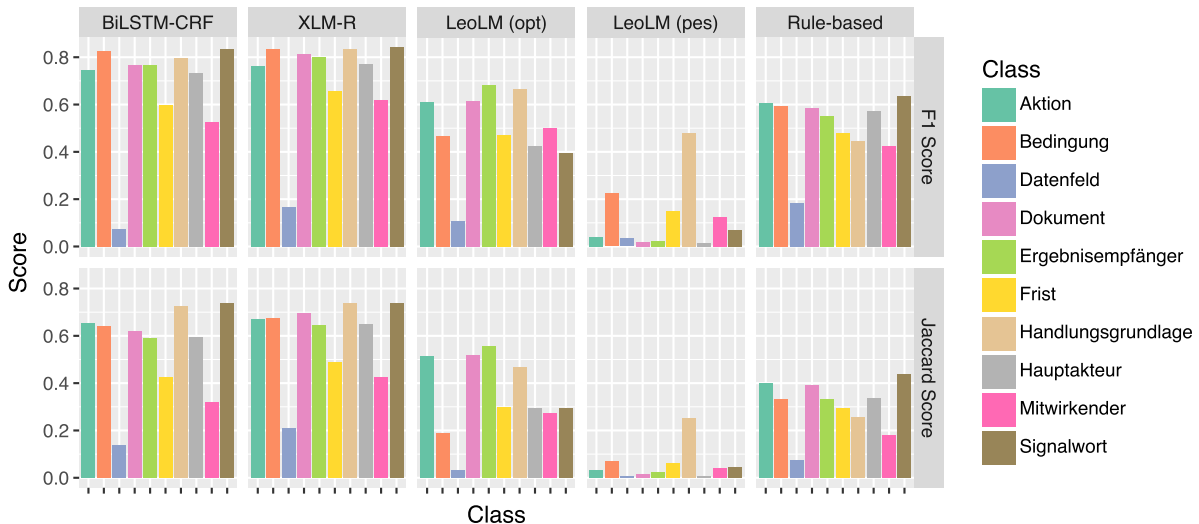


Figure 2: Evaluation results for the different approaches by class and score type.

mentation successfully predicts significantly more test spans belonging to this class than it misses, but the overlap is not that large in many instances. Again, this conclusion is strengthened by the fact that the median value of Jaccard scores is 0.25.

It is interesting to note that *Handlungsgrundlage* ‘legal grounds for action’ has the highest Precision value in the token-based evaluation at 0.78 and *Signalwort* ‘signaling word’ the third highest at 0.73. Therefore it appears that there is a close relation between the Precision value and the entity ranking given by the zero-total ratio. However, the picture is complicated by the fact that the second highest Precision value of 0.75 is attached to the class *Bedingung* ‘condition’. But this class’s zero-total ratio of 0.51 suggests that the system correctly predicts only less than half of the instances of the class while the level of overlap in each case isn’t very high either, as indicated by a relatively low mean value of 0.33. This suggests that this class’s implementation is less-well performing than the token-based evaluation suggests. It further illustrates that although the token-based F-score evaluation and the span-based Jaccard similarity score evaluation roughly point in the same direction, the span-based evaluation allows for a finer-grained interpretation of the system’s performance.

## 6.2 Deep discriminative models

The scores show that the *Datenfeld* ‘data field’ class is the most difficult class to predict, by a large margin. One possible explanation for this could be that it requires much prior knowledge about the actual

process and context to be able to judge whether something classifies as *Datenfeld* ‘data field’ or not. This idea is also supported by the fact that the transformer model solves this task better than the BiLSTM-CRF model (it achieves approx. twice the score). Besides that, the classes *Frist* ‘deadline’ and *Mitwirkender* ‘contributor’ are probably difficult to predict for the models, because even humans struggle to differentiate between *Hauptakteur* ‘main actor’ and *Mitwirkender* ‘contributor’ and *Frist* ‘deadline’ and *Bedingung* ‘condition’ in many cases. This has an additional adverse effect: it raises the probability that the annotations of these edge-cases are inconsistent and thereby makes learning harder for these difficult cases.

For both models the classes *Signalwort* ‘signaling word’ and *Handlungsgrundlage* ‘legal grounds for action’ are among the best-performing. In the latter case this is expected, given that instances of the class *Handlungsgrundlage* ‘legal grounds for action’ are relatively easy to identify based on their structure (e.g. § 44b Absatz 1 Satz ). The class *Signalwort* ‘signaling word’ has a more heterogeneous definition (see Appendix A) which would suggest that classification is more difficult. However, this class has an easily identifiable core in the form of modal verbs or *zu*-infinitive. Apparently, this core is significant enough that recognition proves to be robust.

Regarding the Jaccard Scores it interesting to see that *Handlungsgrundlage* ‘legal grounds for action’ also ranks second place with the BiLSTM-CRF model here even though its F1-score only

ranks 5th. This is presumably because the spans for *Handlungsgrundlage* ‘legal grounds for action’ are generally long. This results in significant token overlap with most reasonable predictions, even if the start and end points are slightly inaccurate.

### 6.3 Deep generative models

In comparison, we see lower values for F1-score for this approach as compared to the work from [Bachinger et al. \(2024\)](#), though in the latter, micro F1-score was used as a measure as compared to macro F1-score here. The best class for both metrics is *Ergebnisempfänger* ‘recipient of service’. The generative approach scores better in general according to the token-based evaluation, which might be due to the fact that the annotation scheme in the prompt is token based.

### 6.4 Comparison

Based on our results, deep discriminative models outperform both the rule-based approach and deep generative models, with the rule-based approach slightly outperforming deep generative models. That deep discriminative models perform well in this task is consistent with other findings in NER research ([Yadav and Bethard, 2019](#); [Pakhale, 2023](#)); but that both rule-based and deep generative approaches only reach the modest scores that we report is surprising, given results in other published research. [Gorinski et al. \(2019\)](#), for instance, compare a rule-based NER system with deep learning and transfer learning systems in the domain of public health records. They found that the hand crafted rule-based system consistently outperforms both the transfer learning and the deep learning systems, reaching an overall F1-score of 0.95. [Wang et al. \(2023\)](#) claim that while unsophisticated applications of LLM to NER perform inferior to supervised learning models, their suggested way of adapting LLMs to the NER task improves performances to state of the art baseline levels.

That both the rule-based approach and the deep generative approach take performance hits at roughly the same scale strongly suggests that there must be a common cause affecting both approaches, rather than individual causes having to do with the implementation of each.<sup>3</sup> The most

<sup>3</sup>This is not to say that there may not be issues with our implementation of these approaches. On the contrary, we are well aware of limitations in the implementation particularly of the rule-based Approach. However, such limitations would only selectively affect one approach and can not explain similarities in outcomes across these approaches.

obvious cause lies in the definition of the classes used in the task. As is apparent from the definitions and examples in [Appendix A](#), the classes are heterogenous with respect to semantic and syntactic types, amalgamating linguistic categorization and legal or administrative classification schemas. [Gorinski et al. \(2019\)](#), in contrast, had linguists designing the classes in close cooperation with domain experts to come up with linguistically motivated entity classes, and [Wang et al. \(2023\)](#) default to the general linguistically motivated entity definitions such as *Location* or *Organization*. It appears that human linguists and general LLMs both have similar difficulties processing heterogeneous classes. deep discriminative models, on the other hand, are able to learn heterogenous class patterns much more easily.

## 7 Conclusion and future work

In this paper, we compared three different approaches for supporting legal norm analysis on German legal texts. We find that the deep discriminative models performed best in 9 out of 10 classes. For class *Datenfeld* ‘data field’, the rule-based approach performed better but the class is in general not well predicted. Future work may explore the integration of these methods, with a promising direction being the combination of deep discriminative approaches and rule-based techniques. Previous research has found this combination to be productive, c.f. for instance the work of [Abdallah et al. \(2012\)](#).

Another intriguing option is the combination of the rule-based approach with the Deep Generative approach. While the latter did not perform as well on the larger dataset compared to the results reported by [Bachinger et al. \(2024\)](#), it stands to reason that the combination with the rule-based approach might improve the performance of both. Since neither the rule-based approach nor the Deep Generative approach requires the retraining of data models, this combination could potentially keep implementation and development costs in a real-world scenario low. One way to combine these approaches would be to automatically extract the  $n$  best matches of the rule-based system for a given class and use these as examples in the prompt for querying an existing LLM.



## 8 Limitations

For the deep discriminative approach, the hyperparameter optimization of both models was limited by the available resources. Without such limitations, the search space for the grid search could have been extended to find better hyperparameters.

For the deep generative approach, there were some limitations due to using a pre existing implementation. The examples in the prompt were predefined and from a smaller data set. They may not be representative of the classes in the overall corpus. Also, the texts used in the related work by (Bachinger et al., 2024) were annotated by different annotators and a different annotation guideline than other parts of the corpus, while maintaining the same classes.

The rule-based system, in part due to early design decisions, can not make use of syntactic information such as dependency relations or phrase structure. Moreover, overlapping annotations and multiple annotations can not be used because we had to stick to the CoNLL 2002 annotation scheme, which does not allow for multiple NER annotations. We plan to address these limitations in the near future.

## References

- Sherief Abdallah, Khaled Shaalan, and Muhammad Shoaib. 2012. [Integrating Rule-Based System with Classification for Arabic Named Entity Recognition](#). In David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, Gerhard Weikum, and Alexander Gelbukh, editors, *Computational Linguistics and Intelligent Text Processing*, volume 7181, pages 311–322. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. [FLAIR: An Easy-to-Use Framework for State-of-the-Art NLP](#). In *NAACL-HLT 2019*, pages 54–59. Association for Computational Linguistics.
- Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. [Contextual String Embeddings for Sequence Labeling](#). In *COLLING 2018*, pages 1638–1649. Association for Computational Linguistics.
- Author Anonymous. 2024. [Gerps-compare: Dataset and code](#).
- Sarah T. Bachinger, Leila Feddoul, Marianne Jana Mauch, and Birgitta König-Ries. 2024. [Extracting legal norm analysis categories from german law texts with large language models](#). In *Proceedings of the 25th Annual International Conference on Digital Government Research*, dg.o ’24, page 481–493, New York, NY, USA. Association for Computing Machinery.
- Dor Bernsohn, Gil Semo, Yaron Vazana, Gila Hayat, Ben Hagag, Joel Niklaus, Rohit Saha, and Kyryl Truskovskiy. 2024. [LegalLens: Leveraging LLMs for legal violation identification in unstructured text](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2129–2145, St. Julian’s, Malta. Association for Computational Linguistics.
- Sergei Bogdanov, Alexandre Constantin, Timothée Bernard, Benoit Crabbé, and Etienne Bernard. 2024. [NuNER: Entity Recognition Encoder Pre-training via LLM-Annotated Data](#). *Preprint*, arXiv:2402.15343.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomáš Mikolov. 2017. [Enriching Word Vectors with Subword Information](#). *Trans. Assoc. Comput. Linguistics*, 5:135–146.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised Cross-lingual Representation Learning at Scale](#). In *ACL 2020*, pages 8440–8451. Association for Computational Linguistics.
- Yongfu Dai, Duanyu Feng, Jimin Huang, Haochen Jia, Qianqian Xie, Yifang Zhang, Weiguang Han, Wei Tian, and Hao Wang. 2023. [Laiw: A chinese legal large language models benchmark \(a technical report\)](#). *arXiv e-prints*, pages arXiv–2310.
- Harshil Darji, Jelena Mitrović, and Michael Granitzer. 2023. [German BERT Model for Legal Named Entity Recognition](#). In *Proceedings of the 15th International Conference on Agents and Artificial Intelligence*, pages 723–728.
- Christopher Dozier, Ravikumar Kondadadi, Marc Light, Arun Vachher, Sriharsha Veeramachaneni, and Ramdev Wudali. 2010. [Named Entity Recognition and Resolution in Legal Text](#). In Enrico Francesconi, Simonetta Montemagni, Wim Peters, and Daniela Tiscornia, editors, *Semantic Processing of Legal Texts: Where the Language of Law Meets the Law of Language*, pages 27–43. Springer, Berlin, Heidelberg.
- Robin Erd, Leila Feddoul, Clara Lachenmaier, and Marianne Jana Mauch. 2022. Evaluation of data augmentation for named entity recognition in the german legal domain. In *AI4LEGAL/KGSum@ISWC*, pages 62–72.
- Leila Feddoul, Sarah T Bachinger, Clara Lachenmaier, Sebastian Apel, Pirmin Karg, Norman Klewer, Denys Forshayt, Robin Erd, and Marianne Mauch. 2024. [Gerps-ner: A dataset for named entity recognition to support public service process creation in germany](#).

- Philip John Gorinski, Honghan Wu, Claire Grover, Richard Tobin, Conn Talbot, Heather Whalley, Cathie Sudlow, William Whiteley, and Beatrice Alex. 2019. [Named Entity Recognition for Electronic Health Records: A Comparison of Rule-based and Machine Learning Approaches](#). *Preprint*, arXiv:1903.03985.
- Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. [IL-TUR: Benchmark for Indian legal text understanding and reasoning](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499, Bangkok, Thailand. Association for Computational Linguistics.
- Sung Jae Jung, Hajung Kim, and Kyoung Sang Jang. 2024. [LLM Based Biological Named Entity Recognition from Scientific Literature](#). In *2024 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 433–435. IEEE.
- Elena Leitner, Georg Rehm, and Julian Moreno-Schneider. 2019. [Fine-Grained Named Entity Recognition in Legal Documents](#). In *Semantic Systems. The Power of AI and Knowledge Graphs*, pages 272–287, Cham. Springer International Publishing.
- Elena Leitner, Georg Rehm, and Julian Moreno-Schneider. 2020. [A Dataset of German Legal Documents for Named Entity Recognition](#). In *LREC 2020*, pages 4478–4485. European Language Resources Association.
- Masoud Monajatipoor, Jiaxin Yang, Joel Stremmel, Melika Emami, Fazlolah Mohaghegh, Mozhdeh Rouhsedaghat, and Kai-Wei Chang. 2024. [LLMs in Biomedicine: A study on clinical Named Entity Recognition](#). *Preprint*, arXiv:2404.07376.
- Marco Naguib, Xavier Tannier, and Aurélie Névéal. 2024. [Few shot clinical entity recognition in three languages: Masked language models outperform LLM prompting](#). *Preprint*, arXiv:2402.12801.
- Kalyani Pakhale. 2023. [Comprehensive Overview of Named Entity Recognition: Models, Domain-Specific Applications and Challenges](#). *Preprint*, arXiv:2309.14084.
- Silvio Peikert, Celia Birle, Jamal Al Qundus, VU Le Duyen Sandra, and Adrian Paschke. 2022. [Extracting references from german legal texts using named entity recognition1](#).
- Björn Plüster. 2023. [Leolm: Igniting german-language llm research](#).
- N Kannaiya Raja, Naol Bakala, and S Suresh. 2019. [NLP: Rule Based Name Entity Recognition](#). *International Journal of Innovative Technology and Exploring Engineering*, 8(11):4285–4290.
- Stefan Schweter and Alan Akbik. 2021. [Flert: Document-level features for named entity recognition](#).
- Amir Soleimani, Christof Monz, and Marcel Worring. 2021. [NLQuAD: A Non-Factoid Long Question Answering Data Set](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1245–1255, Online. Association for Computational Linguistics.
- Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. [GPT-NER: Named Entity Recognition via Large Language Models](#). *Preprint*, arXiv:2304.10428.
- Vikas Yadav and Steven Bethard. 2019. [A Survey on Recent Advances in Named Entity Recognition from Deep Learning models](#). *Preprint*, arXiv:1910.11470.
- Junjie Ye, Nuo Xu, Yikun Wang, Jie Zhou, Qi Zhang, Tao Gui, and Xuanjing Huang. 2024. [LLM-DA: Data Augmentation via Large Language Models for Few-Shot Named Entity Recognition](#). *Preprint*, arXiv:2402.14568.
- Zhen Zhang, Yuhua Zhao, Hang Gao, and Mengting Hu. 2024. [LinkNER: Linking Local Named Entity Recognition Models to Large Language Models using Uncertainty](#). In *Proceedings of the ACM Web Conference 2024*, pages 4047–4058, Singapore Singapore. ACM.
- Jochen Zöllner, Konrad Sperfeld, Christoph Wick, and Roger Labahn. 2021. [Optimizing Small BERTs Trained for German NER](#). *Inf.*, 12(11):443.

## A GerPS-NER dataset classes

**Hauptakteur** ‘main actor’ – The office or person that is mainly responsible for the administration of the service. E.g. *Agentur für Arbeit* ‘Federal Employment Agency’

**Ergebnisempfänger** ‘recipient of service’ – Person or company applying to receive the benefits of the service in question. E.g. *Antragsteller* ‘applicant’

**Mitwirkender** ‘contributor’ – External office or actor that needs to give input at specific points in the administration of the service. E.g. *Deutsches Patent- und Markenamt* ‘German Patent and Trade Mark Office’

**Aktion** ‘action’ – Action carried out by one of the actors in the course of the administration of the service. E.g. *erteilen* ‘to grant’

**Dokument** ‘document’ – Documents that the actors exchange between them. E.g. *Antrag* ‘application form’

**Signalwort** ‘signaling word’ – Word or expression influencing the degree of obligatoriness of a decision made on the basis of this statute. E.g. modal verbs such as *kann* ‘may’; *zu*-Infinitiv *Die Genehmigung ist zu erteilen* ‘permission is to be granted’; adjectives or adverbs such as *angemessen* ‘appropriate’ or *berechtigt* ‘being eligible’; phrases such as *auf Wunsch* ‘if desired’

**Bedingung** ‘condition’ – Preconditions for taking an action. Mostly expressed by conditional clauses.

**Frist** ‘deadline’ – Time limits for certain steps in the administrative process; temporal preconditions. E.g. *spätestens am zehnten Tage vor der Wahl* ‘at the latest on the tenth day before the election’

**Datenfeld** ‘data field’ – Expressions that indicate the content of a data field in a form. E.g. *Vollständige Anschrift* ‘complete address’

**Handlungsgrundlage** ‘legal grounds for action’ – Cross reference to the legal basis for the administrative process in question. E.g. *§3, Absatz 2 des Patentgesetzes* ‘Paragraph 3, section 2 of the patent law’

## B F1-score evaluation results

Class	BiLSTM-CRF			XLM-R			LeoLM (opt)			LeoLM (pes)			Rule-based		
	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R
Action	0.74	0.73	0.76	0.76	0.75	0.77	0.61	0.69	0.55	0.04	0.05	0.04	0.60	0.60	0.61
Condition	0.82	0.80	0.85	0.83	0.85	0.82	0.47	0.57	0.40	0.22	0.29	0.18	0.59	0.75	0.49
Data field	0.07	0.10	0.07	0.17	0.20	0.15	0.11	0.06	0.64	0.03	0.02	0.21	0.18	0.14	0.26
Document	0.77	0.77	0.76	0.81	0.81	0.81	0.61	0.69	0.56	0.02	0.02	0.02	0.59	0.65	0.53
Recipient of service	0.77	0.78	0.76	0.80	0.80	0.80	0.68	0.73	0.64	0.02	0.02	0.02	0.55	0.64	0.49
Deadline	0.60	0.64	0.56	0.66	0.68	0.64	0.47	0.71	0.35	0.15	0.25	0.11	0.48	0.63	0.39
Legal grounds for action	0.80	0.79	0.81	0.84	0.82	0.86	0.66	0.87	0.54	0.48	0.67	0.37	0.44	0.78	0.31
Main actor	0.73	0.70	0.77	0.77	0.76	0.78	0.42	0.37	0.50	0.01	0.01	0.02	0.57	0.57	0.58
Contributor	0.53	0.52	0.54	0.62	0.62	0.62	0.50	0.52	0.49	0.12	0.13	0.12	0.43	0.44	0.41
Signaling word	0.84	0.81	0.86	0.84	0.83	0.86	0.39	0.46	0.34	0.07	0.08	0.06	0.63	0.73	0.56
Micro Average	0.79	0.77	0.81	0.81	0.81	0.81	0.51	0.59	0.45	0.17	0.17	0.17	0.56	0.69	0.47

Table 3: F1, Precision and Recall scores for the evaluated approaches by class and model.

## C Tokens and spans in example annotation

An example of the gold standard annotation of `Corpus/corpus_v2/test/1009.conll` is given in Table 4. Notice that class annotations typically span multiple tokens. This is typically the case in classes that are mostly associated with linguistic expressions at the clause level, such as *Bedingung* ‘condition.’ But also classes which are often expressed by single token spans such as *Signalwort* ‘signaling word’ (see token 8) can at times span multiple tokens, as is the case in this example *im Einvernehmen* ‘with approval’ in tokens 26–27 (indicating that the main actor is not completely free in the determination of the action but must involve another agency as a contributor).

Nr	Token	IOB-class
1	<i>Das</i> ‘the’	O
2	<i>Bundesamt</i> ‘federal office’	B- <i>Hauptakteur</i> ‘main actor’
3	<i>für</i> ‘for’	I- <i>Hauptakteur</i> ‘main actor’
4	<i>Sicherheit</i> ‘security’	I- <i>Hauptakteur</i> ‘main actor’
5	<i>in</i> ‘in’	I- <i>Hauptakteur</i> ‘main actor’
6	<i>der</i> ‘the’	I- <i>Hauptakteur</i> ‘main actor’
7	<i>Informationstechnik</i> ‘information technology’	I- <i>Hauptakteur</i> ‘main actor’
8	<i>kann</i> ‘may’	B- <i>Signalwort</i> ‘signaling word’
9	<i>bei</i> ‘with’	B- <i>Bedingung</i> ‘condition’
10	<i>Mängel</i> ‘shortcomings’	I- <i>Bedingung</i> ‘condition’
11	<i>in</i> ‘in’	I- <i>Bedingung</i> ‘condition’
12	<i>der</i> ‘the’	I- <i>Bedingung</i> ‘condition’
13	<i>Umsetzung</i> ‘implementation’	I- <i>Bedingung</i> ‘condition’
14	<i>der</i> ‘of the’	I- <i>Bedingung</i> ‘condition’
15	<i>Anforderungen</i> ‘requirements’	I- <i>Bedingung</i> ‘condition’
16	<i>nach</i> ‘according to’	I- <i>Bedingung</i> ‘condition’
17	<i>Absatz</i> ‘paragraph’	I- <i>Bedingung</i> ‘condition’
18	<i>Id</i>	I- <i>Bedingung</i> ‘condition’
19	<i>oder</i> ‘or’	O
20	<i>in</i> ‘in’	B- <i>Bedingung</i> ‘condition’
21	<i>den</i> ‘the’	I- <i>Bedingung</i> ‘condition’
22	<i>Nachweisdokumenten</i> ‘proof certificates’	I- <i>Bedingung</i> ‘condition’
23	<i>nach</i> ‘according to’	I- <i>Bedingung</i> ‘condition’
24	<i>Satz</i> ‘sentence’	I- <i>Bedingung</i> ‘condition’
25	<i>I</i>	I- <i>Bedingung</i> ‘condition’
26	<i>im</i> ‘with the’	B- <i>Signalwort</i> ‘signaling word’
27	<i>Einvernehmen</i> ‘approval’	I- <i>Signalwort</i> ‘signaling word’
28	<i>mit</i> ‘of’	O
29	<i>der</i> ‘the’	O
30	<i>Bundesnetzagentur</i> ‘federal network agency’	B- <i>Mitwirkender</i> ‘contributor’
31	<i>die</i> ‘the’	O
32	<i>Beseitigung</i> ‘removal’	O
33	<i>der</i> ‘of the’	O
34	<i>Mängel</i> ‘shortcomings’	O
35	<i>verlangen</i> ‘require’	B- <i>Aktion</i> ‘action’
36	.	O

Table 4: Annotation of the sentence ‘The federal office for security in information technology can in case of shortcomings against the requirements of paragraph 1d or in the proof certificates according to sentence 1 require, with the approval of the federal network agency, the correction of the shortcomings’.